[redacted]

UIUCDCS-R-72-525

AN ALGORITHM FOR THE SOLUTION OF A QUADRATIC EQUATION USING CONTINUED FRACTIONS

by

Kishor Shridharbhai Trivedi

June, 1972

DEPARTMENT OF COMPUTER SCIENCE
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN · URBANA, ILLINOIS

UIUCDCS-R-72-525

AN ALGORITHM FOR THE SOLUTION OF A QUADRATIC EQUATION USING CONTINUED FRACTIONS

by

Kishor Shridharbhai Trivedi

June, 1972

Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, Illinois 61801

*This work was supported in part by the National Science Foundation under Grant No. US NSF GJ-813 and was submitted in partial fulfillment for the Master of Science degree in Computer Science, 1972.

ACKNOWLEDGEMENT

The author wishes to express his sincerest gratitude to his advisor, Professor James E. Robertson. Professor Robertson's patience, guidance, invaluable aid and encouragement, timely suggestions, and continuing interest in this research are most appreciated.

My thanks are also due to colleagues Milos Ercegovac and Lakshmi Goyal for many helpful discussions.

I would also like to thank Mrs. June Wingler for an excellent job of typing this thesis.

TABLE OF CONTENTS

1. INTRODUCTION

Arithmetic methods used in digital systems require that some of the prejudices, that all of us have acquired through our early arithmetic training, shall be overcome. Early designs of arithmetic units of digital computers were largely a mechnaization of the methods used by the human computer. The experience of the last few years has taught us that such efforts, though not always, are generally uneconomic and inefficient. We now illustrate further, the point made here.

First of the human prejudices to be overcome was the use of the decimal number system. An interesting discussion on this point can be found in references [7] and [8], cited at the end of this paper.

The second prejudice was the requirement of uniqueness in the representation of numbers. Over the years, the leaders in the field of digital computer arithmetic have emphasized that the key to fast and efficient arithmetic methods is redundancy in the representation of numbers. Interested readers may see reference [8].

The third prejudice is the use of the positional notation, i.e., a number is thought to be a weighted sum of a series. Unfortunately, the functions that can be easily implemented with this type of representation of numbers are limited to addition (subtraction), multiplication, division and square and higher roots.

DeLugish [3] has presented a continued product formulation that extends the range of easily implemented functions to the logarithm, the

exponential, and the trigonometric and inverse trigonometric functions as well as multiplication, division and square root. He has developed algorithms to evaluate these functions in from one to three "multiplication cycle times."

The above research led us to consider what other representations of numbers exist, and what class of computational procedures can easily be formulated for each such representation. This explains our interest in continued fractions. One must make the distinction that although the computational procedure is based theoretically on a continued fraction formulation, the continued fraction representation is ephemeral as explained in this paper.

2. GENERAL EXPLANATION

A finite continued fraction is represented as follows:

$$\frac{P_k}{Q_k} = \frac{p_1}{q_1} + \frac{p_2}{q_2} + \cdots + \frac{p_k}{q_k}.$$

where P_k and Q_k are determined from the recursions [1]:

$$P_i = q_i P_{i-1} + p_i P_{i-2} \tag{2.1}$$

$$Q_i = q_i Q_{i-1} + p_i Q_{i-2} \qquad i = 2, 3, \ldots, k$$

$$P_0 = 0,\ P_1 = p_1,\ Q_0 = 1,\ Q_1 = q_1$$

It is clear that P_k and Q_k can be separately and simultaneously determined in two binary arithmetic units in (k-1) addition times, if p_i and q_i are chosen to be "simple" in the binary sense. The assumption of $p_i = 1$ for all i has been made in this paper. It is possible to remove this restriction [9].

The choice of a digit set for q_i is governed by the following factors.

(a) It should be "simple" in the binary sense. This means that elements of the set should be powers of 2 (both negative and positive powers are allowed), since then each of the operations (2.1) reduces to a shift and an addition.

(b) The number of elements in the set should be a minimum possible, so that shift hardware is simple.

(c) With the digit set chosen, we get a certain range of infinite continued fractions that are representable. This range should form a continuum over the minimum and maximum limits. This requirement allows us to represent any number in the range as an infinite continued fraction.

(d) Rules of selection of q_i should be both feasible and simple. The significance of this requirement will become clear later.

A two valued digit set {1, 2} was first tried.

If we let u_{max} and u_{min} represent the largest and smallest values, respectively, of the infinite continued fractions representable; then

$$u_{max} = \cfrac{1}{1 + \cfrac{1}{2 + u_{max}}} \qquad \text{or} \qquad u_{max} = (\sqrt{3}-1) \simeq 0.732$$

and

$$u_{min} = \cfrac{1}{2 + \cfrac{1}{1 + u_{min}}} \qquad \text{or} \qquad u_{min} = \frac{1}{2}(\sqrt{3}-1) \simeq 0.366$$

In other words,

$$0.366 \leq \lim_{k \to \infty} \frac{P_k}{Q_k} \leq 0.732$$

Unfortunately, however, $\frac{P_\infty}{Q_\infty}$ does not form a continuum over the range [0.366, 0.732]. In other words, not all numbers in the range can be represented with this digit set.

Proof of this fact is as follows.

Let $\quad 0.366 \leq u \leq 0.732$

and let $\quad u = \dfrac{1}{q_1 + f_1}$

where

$$q_1 \in \{1,2\}$$

and

$$0.366 \leq f_1 \leq 0.732$$

for $\quad q_1 = 1,$

$$0.57737 \leq u \leq 0.732$$

and

for $\quad q_1 = 2,$

$$0.366 \leq u \leq 0.42266$$

Thus the rule of expansion is as follows.

For $\quad 0.366 \leq u \leq 0.42266 \quad$ choose $q_1 = 2$

and

$\quad 0.57737 \leq u \leq 0.732 \quad$ choose $q_1 = 1$

But for $0.42266 < u < 0.57737$, there is no way to choose q_1 from the digit set $\{1,2\}$ with the restriction that f_1 is within the allowable range. This means that, if we do restrict q_1 to the chosen digit set, f_1 will no longer be in the allowable range and therefore cannot be expanded further.

So next we try $q_i \in \{\frac{1}{2}, 1\}$. With this set $u_{max} = 1$ and $u_{min} = \frac{1}{2}$. Furthermore $\frac{P_\infty}{Q_\infty}$ does form a continuum over the range $[\frac{1}{2}, 1]$.

A proof of all these three facts follows.

It is known that with q_i in a given digit set having a maximal digit M and a minimal digit m, the largest infinite c.f. representable is an infinite c.f. of period two, with $q_{2i+1} = m$ and $q_{2i} = M$.

Similarly, the smallest infinite c.f. is an infinite c.f. of period two, with $q_{2i+1} = M$ and $q_{2i} = m$.

With $q_i \in \{\frac{1}{2}, 1\}$, we have $m = \frac{1}{2}$, $M = 1$.

$$u_{max} = \cfrac{1}{\frac{1}{2} + \cfrac{1}{1 + \cfrac{1}{\frac{1}{2} + \cdots}}} = \cfrac{1}{\frac{1}{2} + \cfrac{1}{1 + u_{max}}}$$

Solving which, we get, $u_{max} = 1$. Similarly $u_{min} = \frac{1}{2}$. Thus we have shown that

$$\frac{1}{2} \leq u = \left(\frac{P_\infty}{Q_\infty}\right) \leq 1$$

Next we show the continuum property. Our strategy here is to first show that with the digit set $\{\frac{1}{2}, 1\}$, any number in the range $[\frac{1}{2}, 1]$ can be expanded as a c.f. and then we prove that such a c.f. converges to the given number as the number of terms in the c.f. increases.

Thus we start by giving a method of expansion. Let f_1 be any number such that $\frac{1}{2} \leq f_1 \leq 1$. Now let us expand f_1 as follows.

$$f_1 = \frac{1}{q_1 + f_2}$$

such that $q_1 \in \{\frac{1}{2}, 1\}$.

We also require that the choice of q_1 be made such that $\frac{1}{2} \leq f_2 \leq 1$. Since f_2 is also thought of as a c.f. in the range $[\frac{1}{2}, 1]$ and therefore we can expand f_2 also in the same way and so also for any f_i.

We have the following rules of selection which satisfy these conditions.

If $\quad \frac{1}{2} \leq f_1 < \frac{2}{3} \quad$ choose $q_1 = 1$

If $\quad \frac{2}{3} \leq f_1 \leq 1 \quad$ choose $q_1 = \frac{1}{2}$.

As we said earlier f_i can also be expanded as

$$f_i = \frac{1}{q_i + f_{i+1}}$$

using these rules of selection. When same rules of selection can be applied for any i, we will call such a method of expansion, a consistent method.

We call such a method of expansion, a consistent method.

Thus we can get an expansion of f_1, f_2, ..., f_i, ... f_n to give

$$f_1 = \cfrac{1}{q_1 + \cfrac{1}{q_2 + \cfrac{1}{\ddots + \cfrac{1}{q_n + f_{n+1}}}}} \tag{2.2}$$

Next we prove a Lemma to be used later.

<u>Lemma-1</u>

$$\frac{Q_n}{Q_{n-1}} = q_n + \cfrac{1}{q_{n-1} + \cfrac{1}{q_{n+2} + \ddots + \cfrac{1}{q_1}}}$$

<u>Proof</u>

With the restriction that $p_i = 1$ for all i relations (2.1) become,

$$\begin{aligned} P_i &= q_i \; P_{i-1} + P_{i-2} \\ Q_i &= q_i \; Q_{i-1} + Q_{i-2} \\ P_0 &= 0, \; P_1 = 1, \; Q_0 = 1, \; Q_1 = q_1. \end{aligned} \tag{2.3}$$

Using these relations, we have,

$$\frac{Q_n}{Q_{n-1}} = \frac{q_n \; Q_{n-1} + Q_{n-2}}{Q_{n-1}}$$

$$= q_n + \cfrac{1}{\cfrac{Q_{n-1}}{Q_{n-2}}}$$

$$= q_n + \cfrac{1}{q_{n-1} + \cfrac{1}{\cfrac{Q_{n-2}}{Q_{n-3}}}}$$

$$= q_n + \cfrac{1}{q_{n-1} + \cfrac{1}{q_{n-2} + \ddots + \cfrac{1}{q_1}}}$$

Q.E.D.

Let the given number be $f_1 = \frac{P}{Q}$.

And the finite c.f.

$$\cfrac{1}{q_1 + \cfrac{1}{q_2 + \ddots \cfrac{1}{q_i}}}$$ be denoted by $\frac{P_i}{Q_i}$

where $1 \leq i \leq n$.

Let the error made in approximating $\frac{P}{Q}$ with $\frac{P_i}{Q_i}$ be denoted by δ_i.

For proving the theorem we deal with a general case. Let m be the minimum value of a digit in the digit set of q_i. Let b be the largest number representable with this digit set. That is $b = u_{max}$. And similarly let $a = u_{min}$. Let $a \leq f_1 \leq b$. Now we state and prove the following theorem.

Theorem-1 For a number f_1 in [a,b], if there is a consistent method of expansion of f_1 in the form of a continued fraction, then such an expansion converges to the value f_1 as the number of terms in the expansion increases provided that $m > 0$ and that δ_1 and δ_2 are finite.

Let us assume that f_1 is expanded to the nth term. That is,

$$\frac{P}{Q} = f_1 = \cfrac{1}{q_1 + \cfrac{1}{q_2 + \ddots + \cfrac{1}{q_n + f_{n+1}}}}$$

where $a \leq f_{n+1} \leq b$, since our method of expansion is consistent.

Using relations (2.3) and noting that

$$q_n + f_{n+1} = q_n + \frac{1}{1/f_{n+1}}$$

we have,

$$P = \frac{1}{f_{n+1}} P_n + P_{n-1}$$

$$Q = \frac{1}{f_{n+1}} Q_n + Q_{n-1}$$

$$\frac{P}{Q} = \frac{\frac{1}{f_{n+1}} P_n + P_{n-1}}{\frac{1}{f_{n+1}} Q_n + Q_{n-1}}$$

$$= \frac{\left(\frac{P_n}{Q_n}\right) + \left(\frac{P_{n-1}}{Q_{n-1}}\right) \frac{f_{n+1}\, Q_{n-1}}{Q_n}}{1 + \frac{f_{n+1}\, Q_{n-1}}{Q_n}}$$

let
$$\frac{P}{Q} = \frac{P_n}{Q_n} + \delta_n$$

then
$$\frac{P}{Q} = \frac{\left(\frac{P}{Q} - \delta_n\right) + \left(\frac{P}{Q} - \delta_{n-1}\right) \frac{f_{n+1}\, Q_{n-1}}{Q_n}}{1 + \frac{f_{n+1}\, Q_{n-1}}{Q_n}}$$

$$= \frac{P}{Q} - \frac{\delta_n + \dfrac{\delta_{n-1} f_{n+1} Q_{n-1}}{Q_n}}{1 + f_{n+1} \dfrac{Q_{n-1}}{Q_n}}$$

Which gives

$$\delta_n = -\delta_{n-1} \frac{f_{n+1} Q_{n-1}}{Q_n}$$

Similarly

$$\delta_{n-1} = -\delta_{n-2} \frac{f_n Q_{n-2}}{Q_{n-1}}$$

combining,

$$\delta_n = \delta_{n-2} \frac{f_n f_{n+1} Q_{n-2}}{Q_n}$$

$$= \delta_{n-2} \frac{f_n f_{n+1}}{\dfrac{q_n Q_{n-1} + Q_{n-2}}{Q_{n-2}}}$$

$$= \delta_{n-2} \frac{f_n f_{n+1}}{1 + q_n T_{n-1}} \qquad \text{where} \quad T_{n-1} = \frac{Q_{n-1}}{Q_{n-2}}$$

Also

$$f_n = \frac{1}{q_n + f_{n+1}}$$

$$\therefore \delta_n = \delta_{n-2} \frac{f_n \left(\dfrac{1}{f_n} - q_n \right)}{1 + q_n T_{n-1}}$$

$$= \delta_{n-2} \frac{1 - f_n q_n}{1 + q_n T_{n-1}}$$

let

$$A_n = \frac{1 - f_n q_n}{1 + q_n T_{n-1}}$$

We are interested in showing that $A_n < 1$ for all n or $\max_n (A_n) < 1$.

$$(A_n)_{max} = \frac{1 - (f_n q_n)_{min}}{1 + (q_n)_{min}(T_{n-1})_{min}}$$

$f_n > 0$ {since $q_i > 0 \ \forall i$}

$(q_n)_{min} = m > 0$

By Lemma-1,

$$T_{n-1} = q_{n-1} + \cfrac{1}{q_{n-2} + \ddots}$$

$\therefore (T_{n-1})_{min} > (q_{n-1})_{min} \quad n \geq 2.$

$$\therefore (A_n)_{max} > \frac{1}{1 + m^2} \qquad n \geq 2$$

$$> 1 \qquad \{\text{since } m > 0\}$$

$\therefore \delta_n < \delta_{n-2}$

$\therefore$ To start with if we have δ_1 and δ_2 finite, then obviously as $n \to \infty$, $\delta_n \to 0$ and hence $\frac{P_n}{Q_n} \to \frac{P}{Q}$ Q.E.D.

Now in the case when $q_i \in \{1/2, 1\}$, $m = \frac{1}{2} > 0$ and $1/2 \leq f_1 = \frac{P}{Q} \leq 1$. We have a consistent method of expansion as shown earlier. If we can show that δ_1 and δ_2 are finite, then we can apply Theroem-1 and hence prove the continuum property.

So let us get bounds on δ_1 first.

For $\quad 1/2 \leq \frac{P}{Q} < \frac{2}{3} \quad q_1 = 1 \quad \therefore \frac{P_1}{Q_1} = 1$

$$\frac{1}{3} < \delta_1 \leq \frac{1}{2}$$

for $\quad 2/3 \le \frac{P}{Q} \le 1, \; q_1 = 1/2 \quad \therefore \; \frac{P_1}{Q_1} = 2$

$$\therefore \; 1 \le \delta_1 \le \frac{4}{3}$$

for $\quad 1/2 \le \frac{P}{Q} \le 1, \qquad \frac{1}{3} < \delta_1 \le \frac{4}{3}$

Next we get the bounds on δ_2.

For $\quad \frac{1}{2} \le \frac{P}{Q} \le \frac{3}{5}; \; q_1 = 1, \; q_2 = \frac{1}{2} \quad \therefore \; \frac{P_2}{Q_2} = \frac{1}{3}$

$$\therefore \; -\frac{4}{15} \le \delta_2 \le -\frac{1}{6}$$

for $\quad \frac{3}{5} \le \frac{P}{Q} < \frac{2}{3}; \; q_1 = 1, \; q_2 = 1 \quad \therefore \; \frac{P_2}{Q_2} = \frac{1}{2}$

$$\therefore \; -\frac{1}{6} \le \delta_2 \le -\frac{1}{10}$$

for $\quad \frac{2}{3} \le \frac{P}{Q} \le \frac{6}{7}; \; q_1 = \frac{1}{2}, \; q_2 = \frac{1}{2} \quad \therefore \; \frac{P_2}{Q_2} = \frac{2}{5}$

$$\therefore \; -\frac{16}{35} \le \delta_2 \le -\frac{1}{5}$$

for $\quad \frac{6}{7} < \frac{P}{Q} \le 1; \; q_1 = \frac{1}{2}, \; q_2 = 1 \quad \therefore \; \frac{P_2}{Q_2} = \frac{2}{3}$

$$\therefore \; -\frac{16}{35} \le \delta_2 \le -\frac{1}{10}$$

$\therefore$ for $\quad \frac{1}{2} \le \frac{P}{Q} \le 1; \; -\frac{16}{35} \le \delta_2 \le -\frac{1}{10}.$

Thus we see that δ_1 and δ_2 are finite and thus we have the required result.

3. BASIC RECURSIONS

The particular problem chosen for investigation in this paper was the solution of a limited class of quadratics

$$x^2 + b_k x - c_k = (x-u)\ (x+v) = 0$$

such that $1/2 \leq u \leq 1$. The problem, specifically, is, given b_k and c_k, find u (and hence $v = b_k + u$). This problem was selected because of the following property of infinite periodic continued fractions of period k. If the value of a finite continued fraction formed by truncating the given fraction beyond the $(k-1)^{\underline{st}}$ partial denominator is $\frac{P_{k-1}}{Q_{k-1}}$ and similarly, if the value of a fraction truncated after the k^{th} term be given by $\frac{P_k}{Q_k}$, then the value of the infinite periodic continued fraction is given by u, the positive root of the above quadratic. The coefficients b_k and c_k are, $b_k = (Q_k - P_{k-1})/Q_{k-1}$ and $c_k = P_k/Q_{k-1}$. The problem then, is resolved specifically, to the following one. Given $(Q_k - P_{k-1})/Q_{k-1}$ and P_k/Q_{k-1} (note that k is unknown), find the sequence of partial denominators q_i $(i = 1, 2, \ldots, k)$, and from these by recursions (2.1) form P_n and Q_n to a satisfactory precision. Finally the ratio $\frac{P_n}{Q_n}$ gives the value of u to machine accuracy.

Next we develop the basic recursion* relations for expanding the root u of the quadratic in the form of a continued fraction. Together with recursion relations (2.3), these form the basis of the present investigation.

*Due to J. E. Robertson[2]

From the given quadratic

$$x^2 + b_k x - c_k = 0$$

and substituting u for x, we get

$$u = \frac{c_k}{b_k + u}$$

$$= \frac{c_k}{b_k +} \frac{c_k}{b_k} + \cdot \cdot \cdot \cdot \cdot$$

Thus in the present form u is an infinite continued fraction of period one. However, the partial numerators (c_k) and partial denominators (b_k) are full precision numbers. Therefore, if we attempt to use the recursions (2.1) directly, it would seem that four full precision multiplications and two full precision additions are required at each iterative step. However, with $p_i = c_k$, $q_i = b_k$ $\forall$ i, Robertson[2] has shown that $P_i = Q_{i-1}$ $\forall$ i and thus only one of the recursions (2.1) is required. Even after this reduction of computation, this procedure is clearly not practical.

The strategy now used is that we extend the period of the infinite c.f. for u by one at each iterative step in the following way.

$$\frac{P}{Q} = u = \cfrac{p_1}{q_1 + \cfrac{c_{k-1}}{b_{k-1} + u}}$$

where p_i and q_i are simple in the binary sense, and hence will reduce the full precision multiplication of the direct method to a conditional shift. (Single or may be multiple - right or left.)

From the last equation,

$$q_1 u^2 + (c_{k-1} + b_{k-1} q_1 - p_1)\, u - b_{k-1}\, p_1 = 0.$$

after dividing throughout by q_1 and then equating coefficients with the $u^2 + b_k u - c_k = 0$ we have,

$$c_k = b_{k-1} \frac{p_1}{q_1} \quad \text{or} \quad b_{k-1} = \frac{q_1}{p_1} c_k$$

and
$$c_{k-1} = q_1 b_k + p_1 - \frac{q_1^2 c_k}{p_1}$$

$$c_{k-1} = q_1 (b_k - b_{k-1}) + p_1$$

and
$$b_{k-1} = \frac{q_1 c_k}{p_1} . \qquad (3.4)$$

Let us now take a case when u is already expanded into a periodic fraction of period n. Then

$$\frac{P}{Q} = u = \cfrac{p_1}{q_1 + \cfrac{p_2}{q_2}} + \; . \; . \; . \; + \cfrac{p_n}{q_n + \cfrac{c_{k-n}}{b_{k-n} + u}}$$

With the usual notation for $\frac{P_n}{Q_n}$, $\frac{P_{n-1}}{Q_{n-1}}$ and applying recursions (2.1) to the above equation,

$$\frac{P}{Q} = u = \frac{(b_{k-n}+u)\, P_n + (c_{k-n})\, P_{n-1}}{(b_{k-n}+u)\, Q_n + (c_{k-n})\, Q_{n-1}}$$

from which we obtain the equation,

$$u^2 + u\left(\frac{b_{k-n}Q_n + c_{k-n}Q_{n-1} - P_n}{Q_n}\right) - \frac{c_{k-n}P_{n-1} + b_{k-n}P_n}{Q_n} = 0.$$

Equating coefficients with the given equation

$$u^2 + b_k u - c_k = 0,$$

we get $$P_n b_{k-n} + P_{n-1}c_{k-n} = Q_n c_k$$

and $$Q_n b_{k-n} + Q_{n-1}c_{k-n} = P_n + Q_n b_k$$

Solving for b_{k-n} and c_{k-n},

$$b_{k-n} = \frac{P_{n-1}Q_n b_k + P_n P_{n-1} - Q_n Q_{n-1} c_k}{P_{n-1}Q_n - Q_{n-1}P_n}$$

and $$c_{k-n} = \frac{P_n Q_n b_k + P_n^2 - Q_n^2 c_k}{P_n Q_{n-1} - Q_n P_{n-1}}$$

Now if we put the restriction that $p_i = 1$ for all i then we can use a theorem from the theory of continued fractions[1] which says

$$P_n Q_{n-1} - P_{n-1}Q_n = (-1)^{n-1}$$

$$\text{if } p_i = 1 \quad \forall\, i.$$

Then the last two equations result in

$$b_{k-n} = (-1)^{n-1}\left[Q_n\left(Q_{n-1}c_k - P_{n-1}b_k\right) - P_n P_{n-1}\right] \qquad (3.5)$$

$$c_{k-n} = (-1)^{n-1}\left[Q_n\left(P_n b_k - Q_n c_k\right) + P_n^2\right]. \qquad (3.6)$$

Replacing n by n-1, we have

$$b_{k-n+1} = (-1)^{n-1} [Q_{n-1} (P_{n-2}b_k - Q_{n-2}c_k) + P_{n-2}P_{n-1}] \tag{3.7}$$

$$c_{k-n+1} = (-1)^{n-1} [Q_{n-1} (Q_{n-1}c_k - P_{n-1}b_k) - P^2_{n-1}] \, . \tag{3.8}$$

Similarly,

$$b_{k-n+2} = (-1)^{n-1} [Q_{n-2} (Q_{n-3}c_k - P_{n-3}b_k) - P_{n-3}P_{n-2}] \tag{3.9}$$

$$c_{k-n+2} = (-1)^{n-1} [Q_{n-2} (P_{n-2}b_k - Q_{n-2}c_k) + P^2_{n-2}] \, . \tag{3.10}$$

From these equations, surprisingly enough, we are able to eliminate all P's and Q's and get the relations between b's and c's involving only q's as follows.

Substituting for $Q_n = q_nQ_{n-1} + Q_{n-2}$ and $P_n = q_nP_{n-1} + P_{n-2}$ in (3.5) (and for Q_{n-1}, P_{n-1} and the relations 3.7 to 3.10) and rearranging, we have

$$\begin{aligned} b_{k-n} &= (-1)^{n-1} \{q_nQ_{n-1} (Q_{n-1}c_k - P_{n-1}b_k) - q_nP^2_{n-1} \\ &\qquad + Q_{n-2} (Q_{n-1}c_k - P_{n-1}b_k) - P_{n-2}P_{n-1}\} \\ &= q_nc_{k-n+1} + (-1)^{n-1} [Q_{n-2} (q_{n-1}Q_{n-2}c_k - q_{n-1}P_{n-2}b_k) \\ &\qquad - q_{n-1}P^2_{n-2} + Q_{n-2} (Q_{n-3}c_k - P_{n-3}b_k) - P_{n-3}P_{n-2}] \end{aligned}$$

$$b_{k-n} = q_nc_{k-n+1} - q_{n-1}c_{k-n+2} + b_{k-n+2} \, . \tag{3.11}$$

Similarly,

$$c_{k-n} = q_n \left(b_{k-n+1} - b_{k-n}\right) + c_{k-n+2} \,. \tag{3.12}$$

Thus all in all, we have, given b_k and c_k,

$$b_{k-1} = q_1 c_k$$

$$c_{k-1} = 1 + q_1 \left(b_k - b_{k-1}\right)$$

and for $n = 2, 3, \ldots.$ (3.13)

$$b_{k-n} = q_n c_{k-n+1} - q_{n-1} c_{k-n+2} + b_{k-n+2}$$

$$c_{k-n} = q_n \left(b_{k-n+1} - b_{k-n}\right) + c_{k-n+2}$$

These recursion relations together with relations (2.3) form the core of the algorithm discussed in this paper. We will repeat (2.3)

$$P_0 = 0, \; Q_0 = 1$$

$$P_1 = 1, \; Q_1 = q_1$$

for $n = 2, 3, \ldots$ (2.3)

$$P_n = q_n P_{n-1} + P_{n-2}$$

$$Q_n = q_n Q_{n-1} + Q_{n-2}$$

The algorithm, at this stage, is as follows.

Step 1 Read in values of b_k and c_k. (Assume they are in range.)

Step 2 Initialize $P_0 = 0$, $Q_0 = 1$, $P_1 = 1$

Step 3 From the selection circuit and values of b_k and c_k find q_1.

Step 4 Find $b_{k-1} = q_1 (c_k)$

$$c_{k-1} = 1 + q_1 (b_k - b_{k-1})$$

and $Q_1 = q_1$

set $i = 2$

Step 5 Input b_{k-i+1}, c_{k-i+1} to the selection circuit and get q_i as the output.

Step 6 Find $b_{k-i} = q_i c_{k-i+1} - q_{i-1} c_{k-i+2} + b_{k-i+2}$

$$c_{k-i} = q_i (b_{k-i+1} - b_{k-i}) + c_{k-i+2}$$

$$P_i = q_i P_{i-1} + P_{i-2}$$

$$Q_i = q_i Q_{i-1} + Q_{i-2}$$

Step 7 Is $i > i_{max}$? If no then go to step 5, else go to step 8.

Step 8 Get $u = \frac{P_i}{Q_i}$

The value of i_{max} is determined by the desired precision of the result. We need four binary arithmetic units to compute b_{k-i}, c_{k-i}, P_i, Q_i at each iterative step. Each iterative step consists of pure

shifts and addition (subtraction) operations only. A division must be carried out at the end of the iterative process. We need seven storage registers to store previous values namely, q_{i-1}, b_{k-i+1}, b_{k-i+2}, c_{k-i+1}, c_{k-i+2}, P_{i-1} and Q_{i-1}.

To complete the algorithm, we have to provide for a selection process as required in steps 3 and 5. We develop a selection procedure in the next chapter.

We have placed two restrictions on the set of quadratics, namely, $b_k \geq 0$ and $\frac{1}{2} \leq u \leq 1$. The first restriction is unavoidable, however, the second restriction is really no restriction in the following sense.

Let us assume that for a given quadratic, the above restriction is not satisfied, but $\frac{1}{2} \times 2^j \leq u < 2^j$ is satisfied, where j is an integer. Robertson[2] has suggested a scaling procedure to reduce this problem to the rangeof u required by the algorithm presented in this paper.

The given equation $x^2 + b_k x - c_k = 0$ is multiplied throughout by 2^{-2j}, giving,

$$(2^{-j}x)^2 + (2^{-j}b_k)\ (2^{-j}x) - (2^{-2j}c_k) = 0$$

Substituting $x' = 2^{-j}x$, $b_k' = 2^{-j}b_k$, $c_k' = 2^{-2j}c_k$, we get, $x'^2 + b_k'x' - c_k' = 0 = (x' + v')\ (x' - u')$. And now, clearly, $\frac{1}{2} \leq u' < 1$ is satisfied. Therefore, we can solve for u' starting with b_k' and c_k'. Note that b_k' and c_k' are easily obtained from b_k and c_k by -j and -2j bit shifts respectively. Similarly, the root u is obtained from u' by a j-bit shift.

Secondly, if the algorithm of this paper is to be used for the case $b_k = 0$ (i.e., the square rooting problem) and if we are dealing with floating point numbers, the mantissa of the given c_k will be in the range

$[\frac{1}{2},1]$. But since the exponent can be odd or even, a conditional unnormalizing by one bit shift may be required. Thus, if we can find the square roots of all numbers in the range $[\frac{1}{2},2]$ or in the range $[\frac{1}{4},1]$ we are satisfied. The range $[\frac{1}{4},1]$ was selected, resulting in the range of u given by $[\frac{1}{2},1]$.

Henceforth, we assume that $\frac{1}{2} \leq u \leq 1$ which imples $1/2b_k + 1/4 \leq c_k \leq b_k + 1$, since all the other values of c_k and b_k may be scaled to lie in this range.

A proof that the algorithm presented in this paper converges for the specified conditions will be presented in chapter 5.

4. INVESTIGATION FOR SELECTION RULES

Firstly, let us find the range of c_k and b_k that we can possibly consider with the restriction $\frac{1}{2} \leq u \leq 1$ (imposed by the choice $q_i \in \{\frac{1}{2}, 1\}$).

We have, $c_k = uv$ and $b_k = v - u$. $\therefore$ $c_k = b_k u + u^2$.

For a given value of u, this is a straight line in the (c_k, b_k) plane. Using the maximum and the minimum values of u in turn, we will get two straight lines, namely, $c_k = b_k + 1$ (labelled A in figure 1) and $c_k = \frac{1}{2} b_k + \frac{1}{4}$ (labelled B in figure 1). The area of the (c_k, b_k) plane enclosed by these two lines represents a set of quadratics, for which $\frac{1}{2} \leq u \leq 1$. This area is a double triangular wedge with the vertex P at point $(-\frac{3}{2}, -\frac{1}{2})$ in the plane. This is shown in figure 1.

For the sake of simplicity, let us restrict ourselves to the triangular wedge above and to the right of the vertex.

We have, $u = \dfrac{c_k}{b_k + u}$, which is expanded at the first step, to

$$u = \frac{c_k}{b_k + u} = \frac{1}{q_1 + f_1} \qquad q_1 \in \{\tfrac{1}{2}, 1\} \text{ and where } \frac{1}{2} \leq f_1 = \frac{c_{k-1}}{b_{k-1} + u} \leq 1.$$

The above restriction comes from the fact that f_1 is also an infinite continued fraction and hence must lie within permitted range in our system.

Thus, with $\qquad q_1 = 1/2$ and $1/2 \leq f_1 \leq 1$

$$2/3 \leq u \leq 1$$

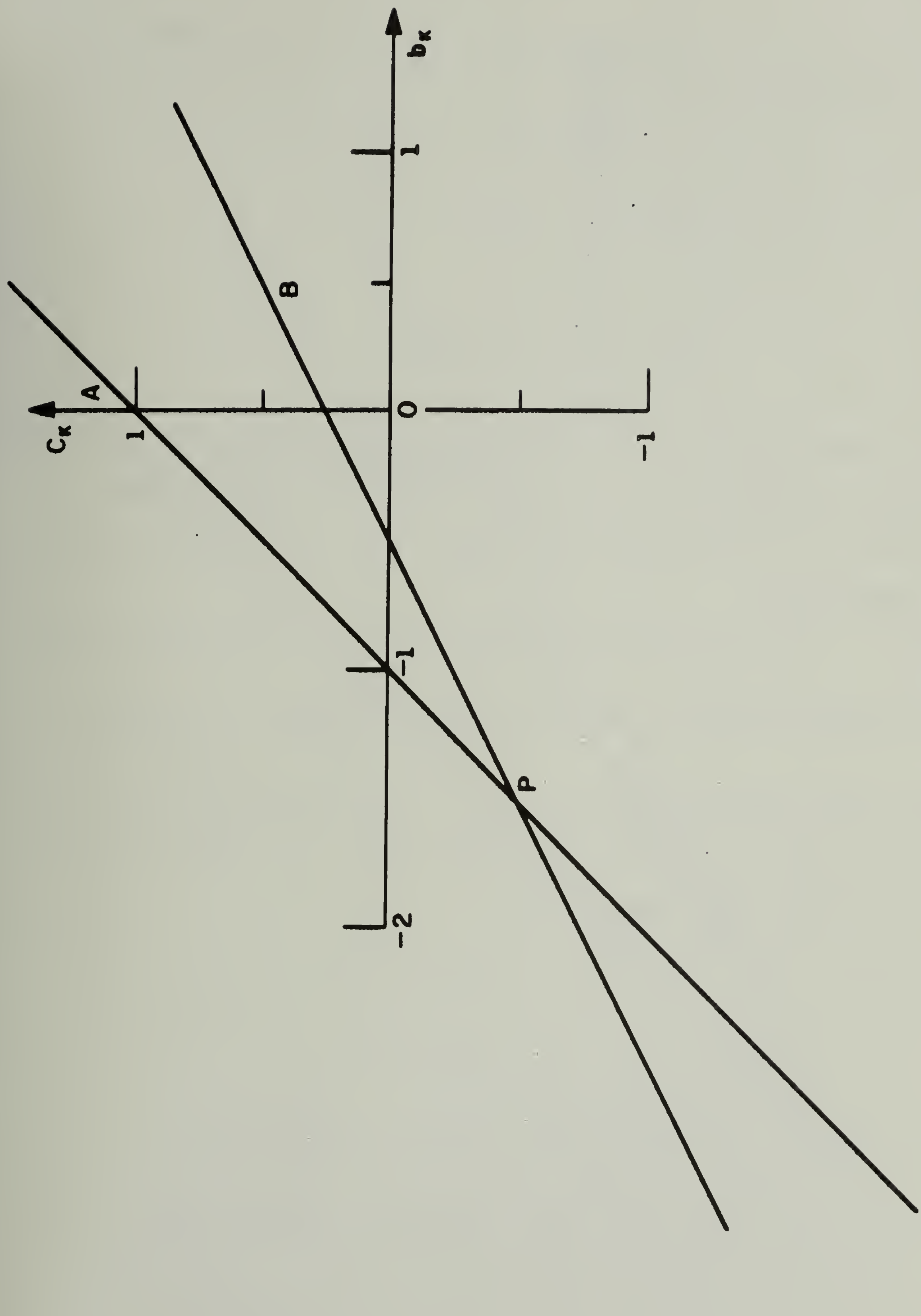

Figure 1

and with $q_1 = 1$ and $1/2 \leq f_1 \leq 1$

$$1/2 \leq u \leq 1$$

or $q_1 = 1/2$ for

$$\frac{2}{3} b_k + \frac{4}{9} \leq c_k \leq b_k + 1$$

and $q_1 = 1$ for

$$\frac{1}{2} b_k + \frac{1}{4} \leq c_k \leq \frac{2}{3} b_k + \frac{4}{9} .$$

Thus given c_k and b_k in the permitted range, we have a procedure to uniquely find q_1. The various areas are shown in figure 2 in which the lines $c_k = b_k + 1$, $c_k = \frac{1}{2} b_k + \frac{1}{4}$ and $c_k = \frac{2}{3} b_k + \frac{4}{9}$ are labelled A, B and C respectively.

For the general case of choice of q_{i+1} $(i \neq 0)$, the procedure is slightly different. Since we have $f_i = \dfrac{c_{k-i}}{b_{k-i} + u} = \dfrac{1}{q_{i+1} + f_{i+1}}$.

Note that except for the case of $i = 0$, f_i is not directly related to u. For the case $i = 0$, we have seen earlier that $f_i = u$. This fact calls for a slightly different approach.

We have $$\frac{1}{2} \leq f_{i+1} \leq 1$$

$$1/2 \leq u \leq 1$$

then $q_{i+1} = \frac{1}{2}$ for $2/3 \leq f_i = \dfrac{c_{k-i}}{b_{k-i} + u} \leq 1$

and $q_{i+1} = 1$ for $1/2 \leq f_i = \dfrac{c_{k-i}}{b_{k-i} + u} \leq 2/3$

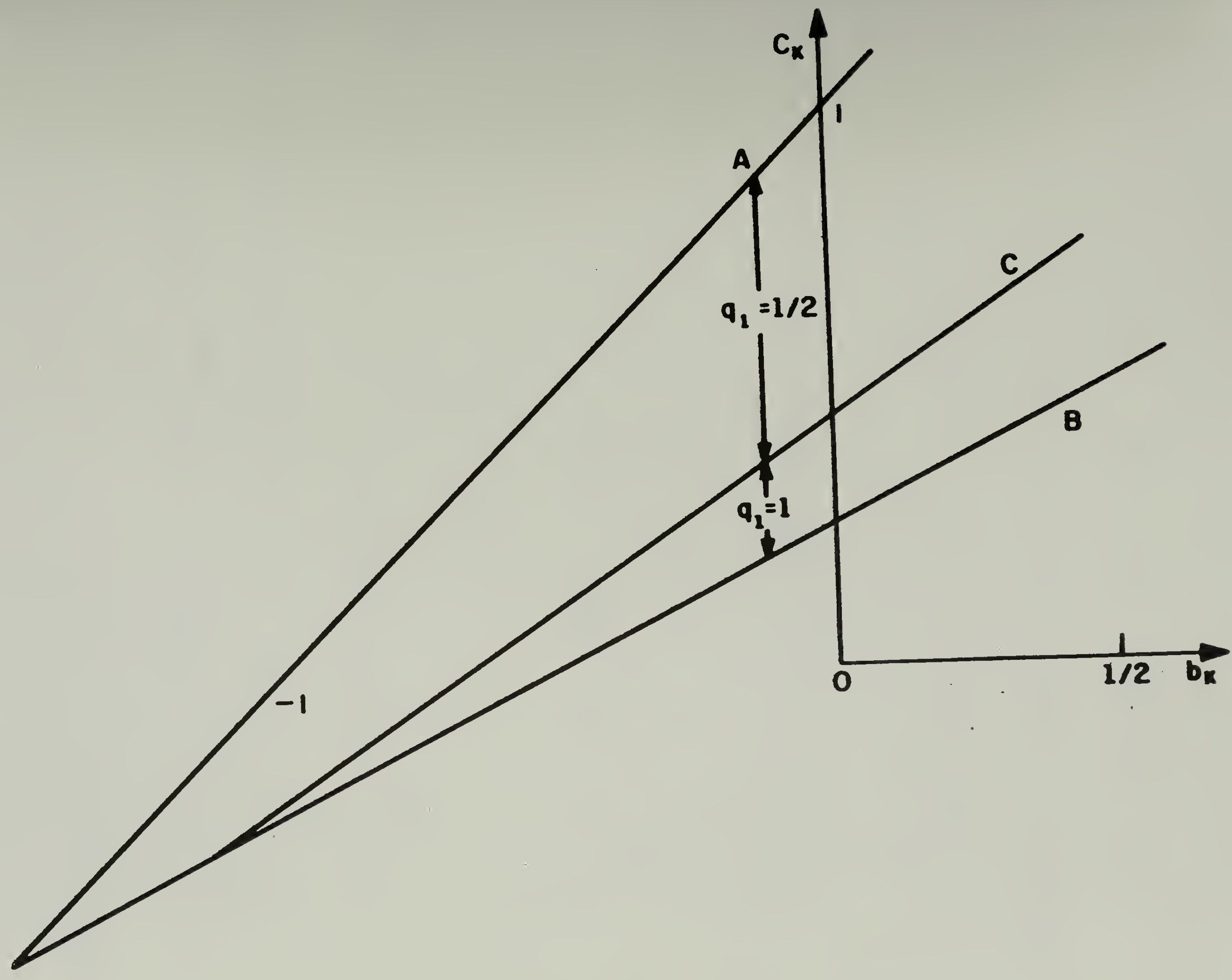

Figure 2

which gives

for $\quad \frac{2}{3} b_{k-i} + \frac{2}{3} u \le c_{k-i} \le b_{k-i} + u; \quad$ choose $q_{i+1} = \frac{1}{2}$

and

for $\quad \frac{1}{2} b_{k-i} + \frac{1}{2} u \le c_{k-i} \le \frac{2}{3} b_{k-i} + \frac{2}{3} u; \quad$ choose $q_{i+1} = 1$.

The dividing line $c_{k-i} = \frac{2}{3} b_{k-i} + \frac{2}{3} u$ decides between $q_{i+1} = 1$ or $q_{i+1} = \frac{1}{2}$. But unfortunately this line is u-dependent, and u is the solution sought; thus, we can never find q_{i+1} unless the solution is known!

Thus we conclude that we must have redundancy in the digit set of q_i, so as to have a greater latitude of choice.

The first choice, naturally enough, was $q_i \in \{\frac{1}{4}, \frac{1}{2}, 1\}$.

Firstly, this set is simple in the "binary sense."

In a manner very similar to the case of $q_i \in \{\frac{1}{2}, 1\}$, it can be shown that

$$a = u_{min} = \frac{1}{8} (\sqrt{17}-1) \simeq 0.390388 \quad \text{and}$$

$$b = u_{max} = \frac{1}{2} (\sqrt{17}-1) \simeq 1.561553.$$

It can also be shown that a consistent method of expansion into a c.f. for a given number in the above range exists.

It is also clear that δ_1 and δ_2 are finite. Since $m = \frac{1}{4} > 0$, theorem 1 is applicable, and therefore, we have the continuum property.

Also, note that the range of u we are interested in, namely [1/2, 1], is completely covered by the range of u allowed by the three-valued digit set.

So let us see if we can find a method of selection of partial denominators.

As before at ith step, we have,

$$f_i = \frac{1}{q_{i+1} + f_{i+1}} \quad \text{where} \quad q_{i+1} \in \{\tfrac{1}{4}, \tfrac{1}{2}, 1\}.$$

We have a constraint on f_{i+1} because of the requirement of consistency of our method. This requirement is that $a = u_{min} \leq f_{i+1} \leq u_{max} = b$. Also note that to start with, $f_1 = u \in [a, b]$. It follows then, that the method of selection that we develop will be applicable for any $i \geq 0$.

We should also be aware of the fact that u is the solution that we seek and hence it is unknown, so we should require that our rules of selection of q_{i+1} be u-independent.

Thus $$f_i = \frac{c_{k-i}}{b_{k-i} + u} = \frac{1}{q_{i+1} + f_{i+1}}$$

$$0.39 \simeq a \leq f_{i+1} \leq b \simeq 1.56$$

with $$q_{i+1} = 1$$

$$0.39 \leq f_i = \frac{c_{k-i}}{b_{k-i} + u} \leq 0.72$$

means

for $0.39\, b_{k-i} + 0.39\, u \leq c_{k-i} \leq 0.72\, b_{k-i} + 0.72\, u$; choose $q_{i+1} = 1$.

Similarly

for $0.485\, b_{k-i} + 0.485\, u \leq c_{k-i} \leq 1.124\, b_{k-i} + 1.124\, u$; choose $q_{i+1} = \frac{1}{2}$

and

for $0.553\, b_{k-i} + 0.553\, u \leq c_{k-i} \leq 1.56\, b_{k-i} + 1.56\, u$; choose $q_{i+1} = \frac{1}{4}$.

The regions, where two choices for q_{i+1} are allowed, are as follows. If

$$0.485\ b_{k-i} + 0.485\ u \leq c_{k-i} \leq 0.72\ b_{k-i} + 0.72\ u, \text{ then}$$

choose $$q_{i+1} = \frac{1}{2} \quad \text{or} \quad 1$$

and if

$$0.553\ b_{k-i} + 0.553\ u \leq c_{k-i} \leq 1.124\ b_{k-i} + 1.124\ u, \text{ then}$$

choose $$q_{i+1} = \frac{1}{4} \quad \text{or} \quad \frac{1}{2}.$$

Let us call the former overlap region the $(\frac{1}{2}\ \&\ 1)$ region, and the latter, the $(\frac{1}{4}\ \&\ \frac{1}{2})$ region. Notice that except for these two regions, the value of q_{i+1} is unique.

A selection line which decides between the choice of $q_{i+1} = \frac{1}{2}$ or 1 should completely lie within the $(\frac{1}{2}\ \&\ 1)$ region. Similarly there is a $(\frac{1}{4}\ \&\ \frac{1}{2})$ selection line. We also require that the selection lines be u-independent, and that the slope as well as the intercept on the c_{k-i} axis be "simple" binary numbers. To do this, we take the intersection of all $(\frac{1}{2}\ \&\ 1)$ regions as u varies over $[a, b]$, and we require that our selection line be completely within this intersection. Similar constraints are necessary for the $(\frac{1}{4}\ \&\ \frac{1}{2})$ selection line. These two intersection regions are shown in figure 3. The upper bound and the lower bound of the $(\frac{1}{4}\ \&\ \frac{1}{2})$ are labelled A and B respectively, and the corresponding bounds of the $(\frac{1}{2}\ \&\ 1)$ region are labelled C and D respectively. The largest upper bound on c_{k-i}, namely $c_{k-i} = 1.56\ b_{k-i} + 1.56\ u_{max}$ (labelled H) and the least lower bound, namely $c_{k-i} = 0.39\ b_{k-i} + 0.39\ u_{min}$ (labelled L) are also shown in figure 3.

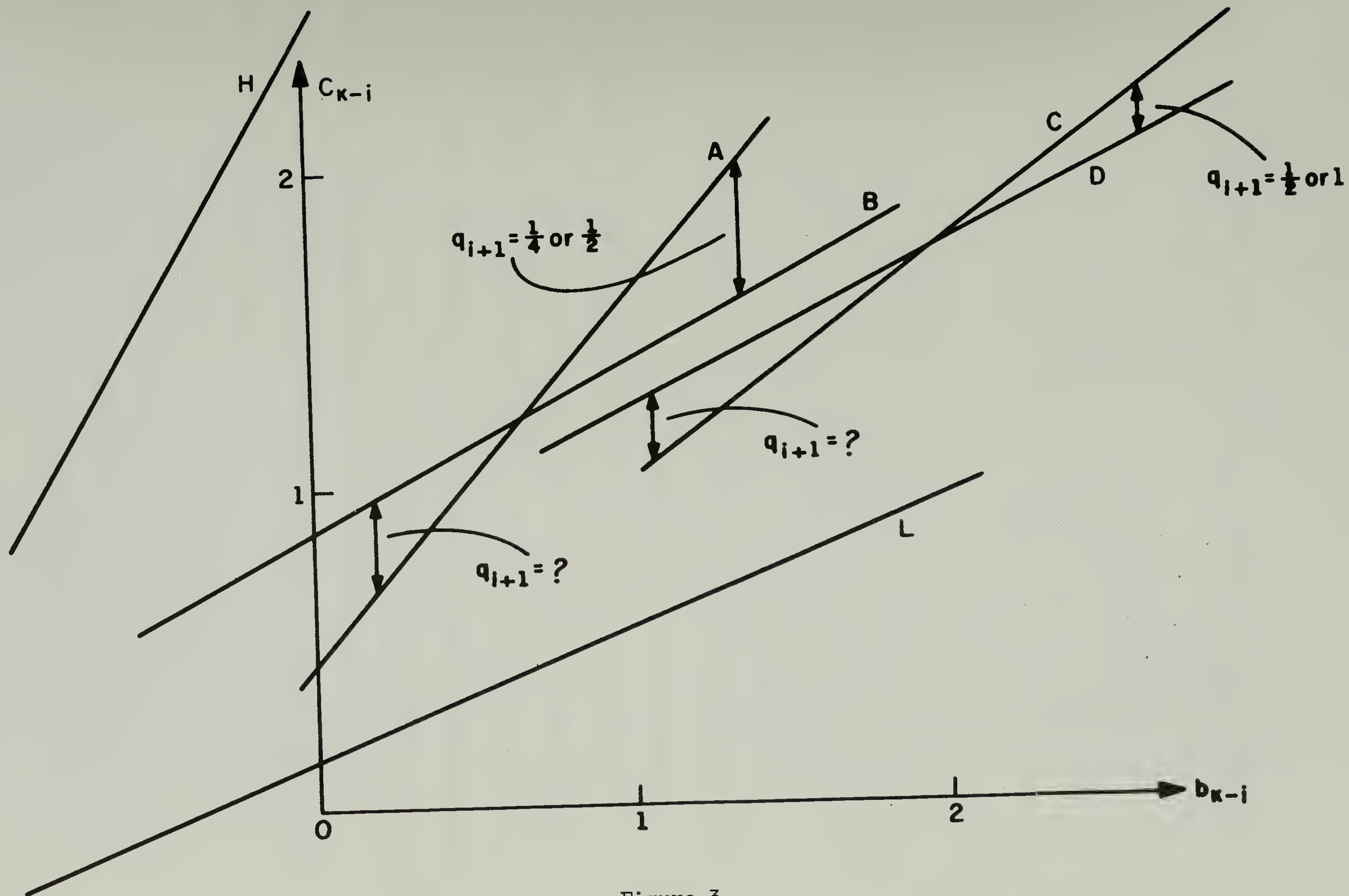
H
c_{K-i}
2
1
0
1
2
b_{K-i}
A
B
C
D
L
$q_{i+1} = \frac{1}{4}$ or $\frac{1}{2}$
$q_{i+1} = \frac{1}{2}$ or 1
$q_{i+1} = ?$
$q_{i+1} = ?$

Figure 3

It is apparent from figure 3, that the choice of q_{i+1} cannot be made (independent of u) for a good part of the (c_{k-i}, b_{k-i}) plane. In particular for $i = 0$ and the case of the square rooting problem, we do not have a choice.

It is conceivable to break up the range of u (i.e., [a, b]) into several parts and then obtain the selection lines, since in that case the intersection of all $(\frac{1}{2}$ &1) regions, when u varies over a smaller range, will be clearly larger.

As we started out to solve the problem for $1/2 \leq u \leq 1$, it seems natural to restrict the range of u to [1/2, 1]. But even then, as it turns out, satisfactory overlap regions cannot be formed.

Dividing the range of u into 2 parts, namely $[\frac{1}{2}, \frac{1}{\sqrt{2}})$ and $[\frac{1}{\sqrt{2}}, 1]$, was chosen first; but this had two disadvantages. Firstly, since the selection procedure for the two ranges will be different, we must know, at the outset, which u-range we are in. This means given c_k, b_k, we should compare $c_k - \frac{1}{\sqrt{2}} b_k$ with $\frac{1}{2}$. Except for $b_k = 0$ (i.e., the square-rooting problem), this is not convenient and is inexact in any finite precision machine. Secondly it was very difficult to show that the resulting algorithm is convergent.

It was then decided to break up the u-range into 3 sections, namely $I_1 = [1/2, 5/8)$, and $I_3 = [5/8, 3/4)$, and I_3 [3/4, 1]. Given the values of c_k and b_k, the following rules are used to decide the range of u.

(1)
$$\left.\begin{array}{l} c_k - \frac{1}{2} b_k \geq \frac{1}{4} \\ \text{and} \\ c_k - \frac{5}{8} b_k < \frac{25}{64} \end{array}\right\} \Rightarrow u \in I_1$$

$$\left.\begin{array}{ll} (2) & c_k - \frac{5}{8} b_k \geq \frac{25}{64} \\ \text{and} & \\ & c_k - \frac{3}{4} b_k < \frac{9}{16} \end{array}\right\} \Rightarrow u \in I_2$$

$$\left.\begin{array}{ll} (3) & c_k - \frac{3}{4} b_k \geq \frac{9}{16} \\ \text{and} & \\ & c_k - b_k \leq 1 \end{array}\right\} \Rightarrow u \in I_3$$

We now develop the rules of selection for all three u-ranges. For the first range, $I_1 = [1/2, 5/8)$, the ($\frac{1}{2}$ & 1) region is given by $0.485\ (b_{k-i} + u) \leq c_{k-i} \leq 0.72\ (b_{k-i} + u)$. Taking the intersection of all these regions as u varies over the range I_1, we obtain; for $0.485\ (b_{k-i} + 5/8) \leq c_{k-i} \leq 0.72\ (b_{k-i} + \frac{1}{2})$; $q_{i+1} = \frac{1}{2}$ or 1. Similarly the intersection of all ($\frac{1}{4}$ & $\frac{1}{2}$) regions, is; for $0.553\ (b_{k-i} + 5/8) \leq c_{k-i} \leq 1.12\ (b_{k-i} + \frac{1}{2})$; $q_{i+1} = \frac{1}{4}$ or $\frac{1}{2}$. These overlap regions are shown in figure 4. This figure needs some explanation.

We know that for any i, $c_{k-i} \leq 1.56\ (b_{k-i} + u)$. Therefore the largest upper bound on c_{k-i} is given by $c_{k-i} = 1.56\ (b_{k-i} + 5/8)$ for this range. We call this line H. Similarly the least lower bound on c_{k-i} is given by $c_{k-i} = 0.39\ (b_{k-i} + \frac{1}{2})$; we call this line L. We call the upper bound of ($\frac{1}{4}$ & $\frac{1}{2}$) overlap region, A; and its lower bound B. We call the upper bound of (1/2 & 1) overlap region, C and its lower bound D.

Note that the region below line A and above line B is the ($\frac{1}{4}$ & $\frac{1}{2}$) region, or in other words, q_{i+1} can be chosen as $\frac{1}{4}$ or $\frac{1}{2}$ in this area for any value of $u \in I_1$. But the area above line A and below line B is a

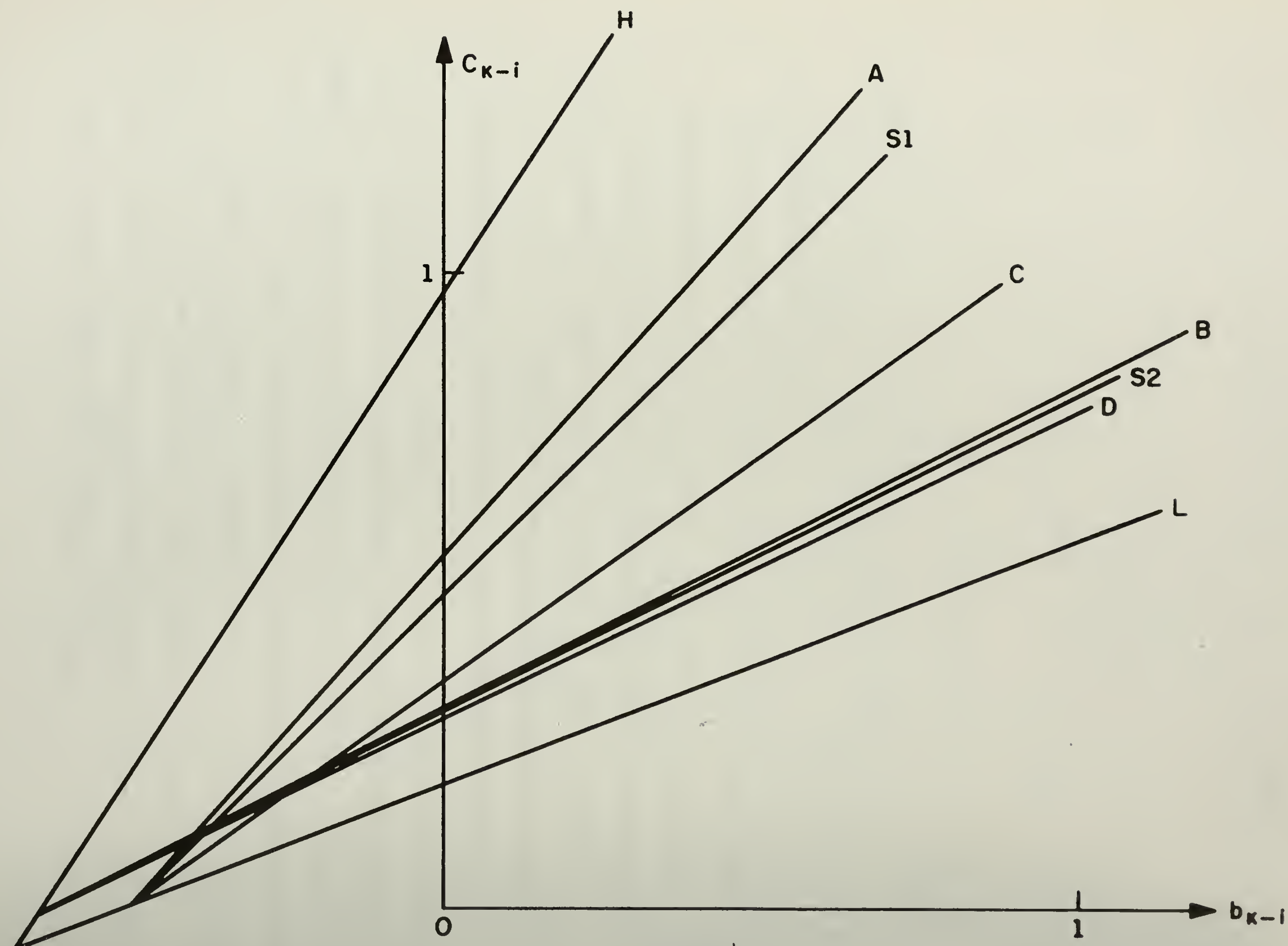

Figure 4

forbidden region, since a choice of q_{i+1} independent of the value of u cannot be made. A similar observation can be made about the $(\frac{1}{2}$ & $1)$ region and thus we also have the $(\frac{1}{2}$ & $1)$ forbidden region. We should now find two selection lines corresponding to these two overlap regions so as to make the choice of q_{i+1} unique. Clearly, each selection line must lie within the respective overlap region, its slope and its intercept on the c_{k-i} axis must both be "simple" binary numbers and it should pass through the vertex of the overlap triangular region as closely as possible.

We chose line S1 in the $(\frac{1}{4}$ & $\frac{1}{2})$ region to be $c_{k-i} = b_{k-i} + \frac{1}{2}$ and line S2 in the $(\frac{1}{2}$ & $1)$ region to be $c_{k-i} = \frac{1}{2} b_{k-i} + \frac{5}{16}$.

Thus the following tests need to be made to determine q_{i+1} for any $u \in I_1$

(1) If $c_{k-i} \leq \frac{1}{2} b_{k-i} + \frac{5}{16}$ then $q_{i+1} = 1$

(2) If $c_{k-i} > b_{k-i} + \frac{1}{2}$ then $q_{i+1} = \frac{1}{4}$

(3) Otherwise $q_{i+1} = \frac{1}{2}$.

Notice that with the selection lines S1 and S2 the forbidden regions are slightly enlarged since the selection lines do not exactly pass through the vertex of the respective overlap triangle.

Thus the $(\frac{1}{4}$ & $\frac{1}{2})$ forbidden region, is, the area enclosed by lines H, B, S1, L. That is, to the left of the intersection point of lines B and S1. Similarly, the $(\frac{1}{2}$ & $1)$ forbidden region is enclosed by lines H, S2, C, L.

Thus we now have a selection procedure for choosing q_{i+1} when c_{k-i} and b_{k-i} are within lines H and L, provided that they do not fall in the forbidden region. Starting from the first step, if we can make

sure that at every subsequent step, (c_{k-i}, b_{k-i}) stay within this permitted region, the requirement of consistency of the selection procedure will then be satisfied.

From the range restriction imposed on f_{i+1}, at the beginning of this chapter it is easily verified that (c_{k-i-1}, b_{k-i-1}) are kept between lines H and L. In the next chapter we also show that we never go into the forbidden region. This we do for all the three u-ranges.

For the second range $I_2 = [5/8, 3/4)$, the intersection of all $(\frac{1}{2}$ & $1)$ regions is given by $0.485\ (b_{k-i} + 3/4) \leq c_{k-i} \leq 0.72\ (b_{k-i} + 5/8)$. Similarly the intersection of all $(\frac{1}{4}$ & $\frac{1}{2})$ regions, is, $0.553\ (b_{k-i} + 3/4) \leq c_{k-i} \leq 1.12\ (b_{k-i} + 5/8)$.

These overlap regions are shown in figure 5. Labelling of lines is similar to the range I_1. Thus, line H is, $c_{k-i} = 1.56\ (b_{k-i} + 3/4)$. Line L is, $c_{k-i} = 0.39\ (b_{k-i} + 5/8)$. We choose selection line S1 as $c_{k-i} = b_{k-i} + 5/8$ and selection line S2 as $c_{k-i} = \frac{1}{2} b_{k-i} + \frac{3}{8}$.

As before, the $(\frac{1}{4}$ & $\frac{1}{2})$ forbidden region is to the left of the point of intersection of lines S1 and B, and is enclosed by lines H, B, S1 and L. The $(\frac{1}{2}$ & $1)$ forbidden region is to the left of the point of intersection of lines S2 and C, and is enclosed by lines H, S2, C and L.

For the third range $I_3 = [3/4, 1]$, the intersection of all $(\frac{1}{2}$ & $1)$ regions is given by, $0.485\ (b_{k-i} + 1) \leq c_{k-i} \leq 0.72\ (b_{k-i} + 3/4)$. Similarly the intersection of all $(\frac{1}{4}$ & $\frac{1}{2})$ regions, is, $0.553\ (b_{k-i} + 1) \leq c_{k-i} \leq 1.12\ (b_{k-i} + 3/4)$.

These overlap regions are shown in figure 6. Labelling of lines is similar to the previous ranges. Thus line H is, $c_{k-i} = 1.56\ (b_{k-i} + 1)$. Line L is, $c_{k-i} = 0.39\ (b_{k-i} + 3/4)$. We choose the selection line S1 as

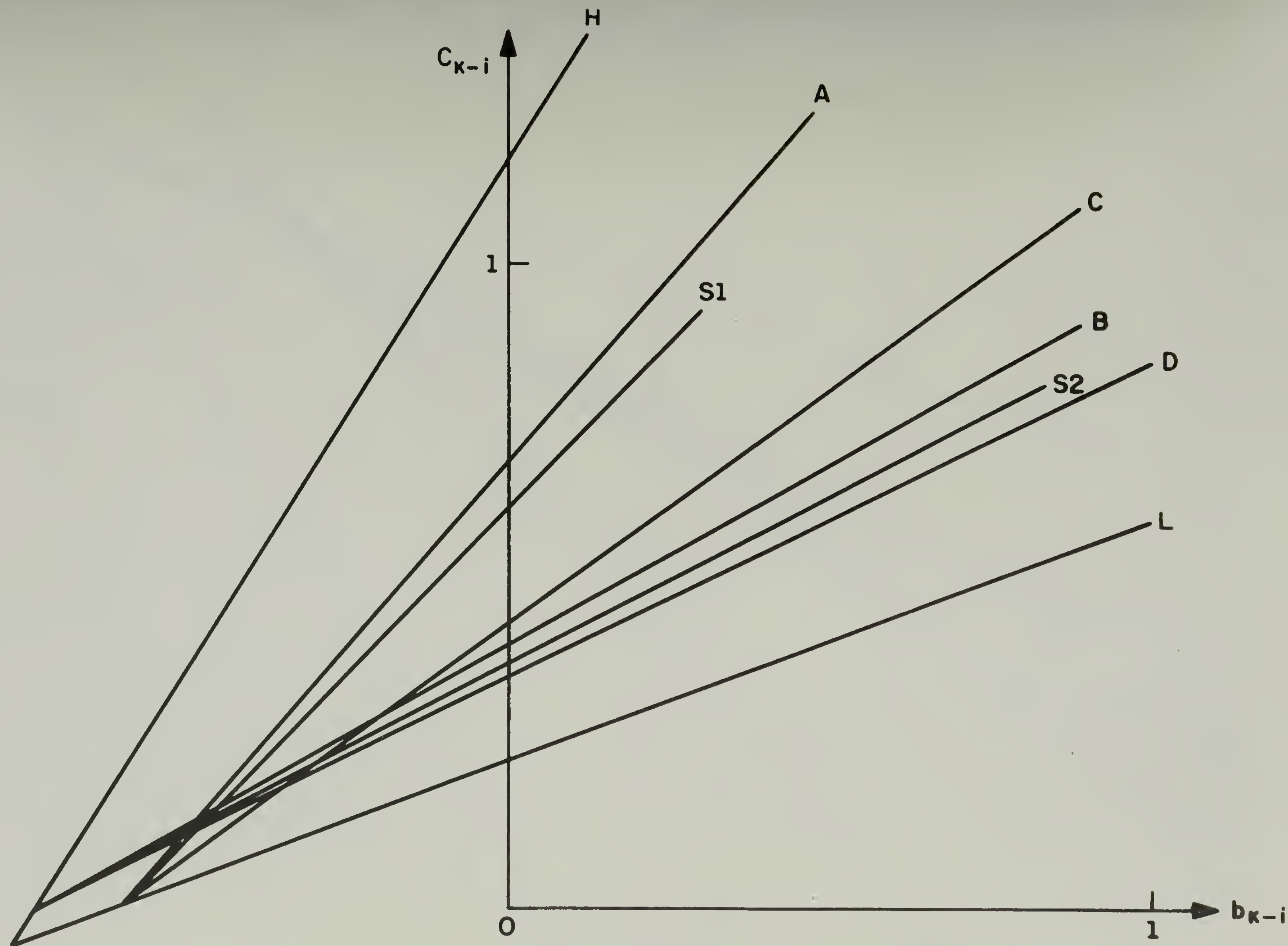

Figure 5

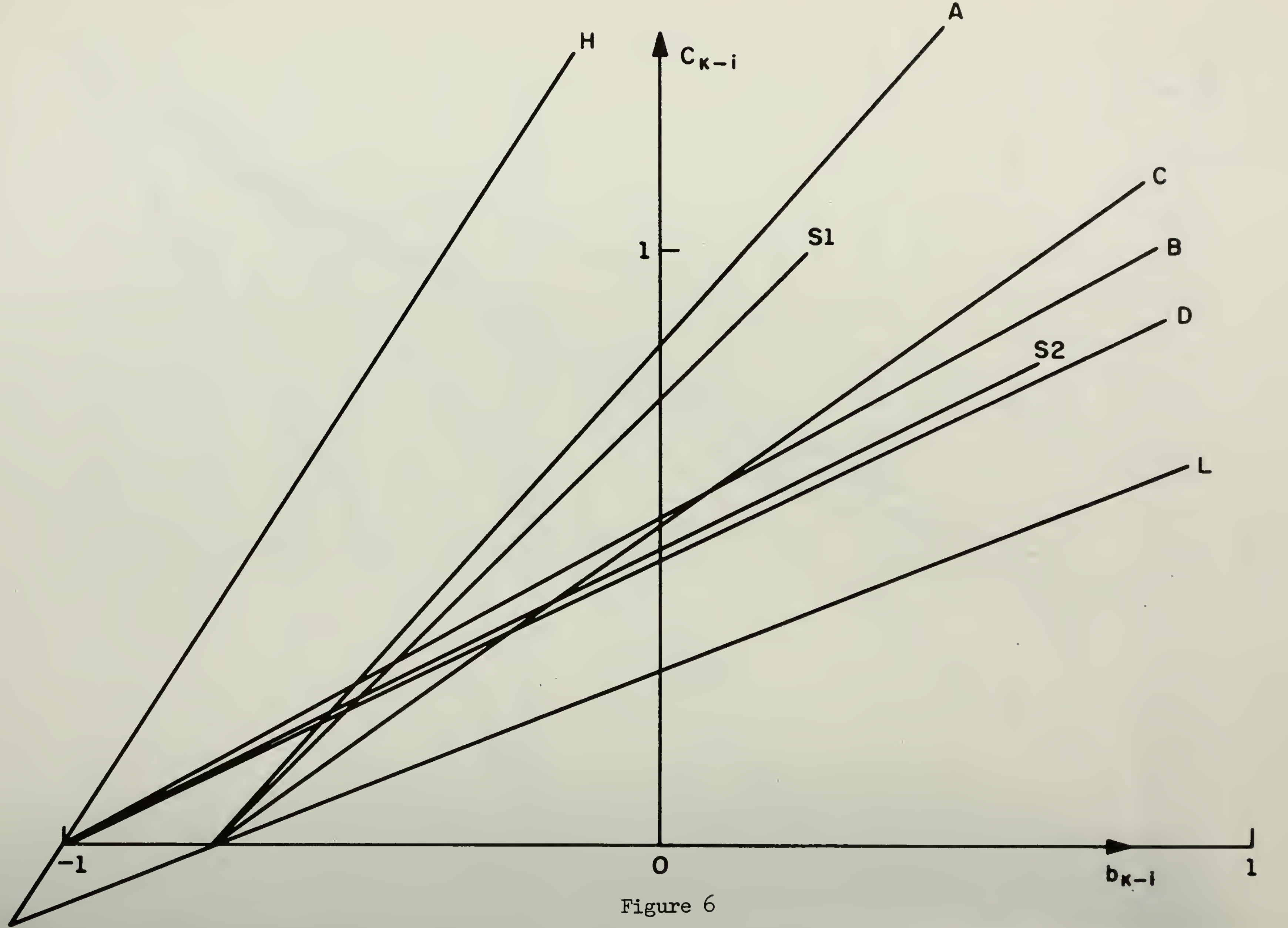

Figure 6

$c_{k-i} = b_{k-i} + \frac{3}{4}$ and selection line S2 as $c_{k-i} = \frac{1}{2} b_{k-i} + \frac{1}{2}$. Forbidden regions are as in previous ranges.

Although this general selection procedure is applicable for all $i \geq 0$, we want to use a special procedure for the case $i = 0$. This is because initial tests are necessary to decide the range of u. We want to use these same tests for deciding q_1.

We have observed earlier that for $i = 0$, $f_i = u$. Then from our previous analysis,

for $q_1 = 1/2$

$$0.485 \leq f_0 = u \leq 1.124$$

and for $q_1 = 1$

$$0.39 \leq f_0 = u \leq 0.72.$$

Then for $u \in I_1$ we choose $q_1 = 1$ and for $u \in I_2$ or I_3 we choose $q_1 = 1/2$. Notice that for all three u-ranges the selection line S1 is of the form $c_{k-i} = b_{k-i} + k1$. Thus the slope of S1 does not change with the range, only the intercept on the c_{k-i} axis changes. Similarly line S2 is of the form $c_{k-i} = \frac{1}{2} b_{k-i} + k2$. We take advantage of this fact in writing the complete algorithm that follows. In the first step of the algorithm, while deciding the range of u, we set two switching variables J_1 and J_2 as appropriate. These two switching variables later decide k1 and k2.

We now give the complete algorithm, which solves a quadratic equation $x^2 + b_k x - c_k = 0$ with $b_k \geq 0$ and $(c_k - b_k \leq 1,\ c_k - \frac{1}{2} b_k \geq \frac{1}{4})$ and gives us the positive root, u, of the above equation.

ALGORITHM QD:

QD-0: [check] If $b_k \leq 0$ then exit, no solution.

If $(c_k - \frac{1}{2} b_k) < \frac{1}{4}$ or if

$(c_k - b_k) > 1$ then exit, no solution.

QD-1: [range] Now set $J_1 \leftarrow J_2 \leftarrow 0$;

If $c_k - \frac{5}{8} b_k < \frac{25}{64}$ then set $q_1 \leftarrow 1$

and go to step QD-2;

set $q_1 \leftarrow \frac{1}{2}$;

If $c_k - \frac{3}{4} b_k < \frac{9}{16}$ then set $J_1 \leftarrow 1$ and

go to step QD-2;

otherwise set $J_2 \leftarrow 1$;

QD-2: [Init] Set $P_0 \leftarrow 0$, $Q_0 \leftarrow 1$, $P_1 \leftarrow 1$, $Q_1 \leftarrow q_1$;

QD-2: [ ] Set $b_{k-1} \leftarrow q_1 c_k$

$c_{k-1} \leftarrow 1 + q_1(b_k - b_{k-1})$

$i \leftarrow 2$

QD-4: [select] If $c_{k-i+1} > (b_{k-i+1} + \frac{1}{2} + \frac{J_1}{8} + \frac{J_2}{4})$ then

set $q_i \leftarrow \frac{1}{4}$ and go to step QD-5;

If $c_{k-i+1} \leq (\frac{1}{2} b_{k-i+1} + \frac{5}{16} + \frac{J_1}{16} + \frac{3J_2}{16})$ then

set $q_i \leftarrow 1$ and go step QD-5;

otherwise set $q_i \leftarrow \frac{1}{2}$;

QD-5: [advance]

$b_{k-i} \leftarrow q_i c_{k-i+1} - q_{i-1} c_{k-i+2} + b_{k-i+2}$

$c_{k-i} \leftarrow q_i(b_{k-i+1} - b_{k-i}) + c_{k-i+2}$

$P_i \leftarrow q_i P_{i-1} + P_{i-2}$

$Q_i \leftarrow q_i Q_{i-1} + Q_{i-2}$

$i \leftarrow i + 1$

QD-6: [loop test] If $i \leq i_{max}$ then go to step QD-4;

QD-7: [final] u (=ROOT) $\leftarrow P_i/Q_i$.

The value of i_{max} will be determined by the desired precision of the result in case this algorithm is implemented in hardware. If this algorithm is implemented in software, however, the value of i_{max} will be determined by the allowable error.

5. PROOF OF CONVERGENCE

We have given algorithm QD in the previous chapter. In this chapter we prove that algorithm QD is convergent. (For the conditions specified in that algorithm, i.e., $b_k \geq 0$, and $c_k - \frac{1}{2} b_k \geq \frac{1}{4}$ and $c_k - b_k \leq 1$.)

Our strategy will be, first to prove that the rules of selection given in algorithm QD are consistent. Then using theorem-1 (chapter 2) we show convergence. We also show that the residual approaches zero as the number of iterations increases.

We first prove a lemma to be used later.

Lemma 2 The following results[2] hold for n > 0.

$$\left(\frac{P_{n-1}}{Q_{n-1}}\right)\left(\frac{P_n}{Q_n}\right) + \left(\frac{P_{n-1}}{Q_{n-1}}\right) b_k - c_k = (-1)^n \frac{b_{k-n}}{Q_n Q_{n-1}} \tag{5.1}$$

$$\left(\frac{P_n}{Q_n}\right)^2 + \left(\frac{P_n}{Q_n}\right) b_k - c_k = (-1)^{n+1} \frac{c_{k-n}}{Q_n^{\,2}} \tag{5.2}$$

$$c_{k-n} = \frac{P_n}{Q_{n-1}} - \frac{Q_n}{Q_{n-1}} (b_{k-n} - b_k) \tag{5.3}$$

Proof Consider the expansion of u up to q_n.

$$u = \cfrac{}{q_1 + \cfrac{1}{q_2 + \cdots \cfrac{1}{q_n + \cfrac{c_{k-n}}{b_{k-n} + u}}}}$$

Consider c_{k-n} to be $(n + 1)^{st}$ partial numerator and $(b_{k-n} + u)$ to be $(n + 1)^{st}$ partial denominator and use standard recursions (2.1) to get

$$P'_{n+1} = (b_{k-n} + u)P_n + c_{k-n}P_{n-1}$$

and

$$Q'_{n+1} = (b_{k-n} + u)Q_n + c_{k-n}Q_{n-1}$$

also

$$u = \frac{P'_{n+1}}{Q'_{n+1}} = \frac{(b_{k-n} + u)P_n + c_{k-n}P_{n-1}}{(b_{k-n} + u)Q_n + c_{k-n}Q_{n-1}}$$

which gives the equation

$$u^2 + u\,\frac{b_{k-n}Q_n + c_{k-n}Q_{n-1} - P_n}{Q_n} - \frac{b_{k-n}P_n + c_{k-n}P_{n-1}}{Q_n} = 0.$$

Recall that u is the root of the quadratic $x^2 + xb_k - c_k = 0$. Therefore comparing coefficients, we get

$$b_k = \frac{b_{k-n}Q_n + c_{k-n}Q_{n-1} - P_n}{Q_n} \tag{5.4}$$

and

$$c_k = \frac{b_{k-n}P_n + c_{k-n}P_{n-1}}{Q_n}\,. \tag{5.5}$$

By rearranging (5.4) we get equation (5.3). By eliminating c_{k-n} between equations (5.3) and (5.4) and then using the identity $P_nQ_{n-1} - Q_nP_{n-1} = (-1)^{n+1}$ we get equation (5.1). Similarly by eliminating b_{k-n} between equations (5.3) and (5.4) and again using the above identity, we get equation (5.2). Q.E.D.

Now we prove that the rules of selection of partial denominators are consistent. For this, we first prove that for all $i \geq 0$, the points

(c_{k-i}, b_{k-i}) remain within the area enclosed by lines H and L in figures 4, 5 or 6 as the case may be, depending the u-range under consideration. For $i = 0$, this is quite clearly true. For all $i \geq 0$, we had put a restriction on f_{i+1}, namely,

$$0.39 \leq f_{i+1} = \frac{c_{k-i-1}}{b_{k-i-1} + u} \leq 1.56,$$

which may be written as,

$$0.39 \, (b_{k-i-1} + u) \leq c_{k-i-1} \leq 1.56 \, (b_{k-i-1} + u),$$

which clearly shows that we are always within the limits of lines H and L. Next, we have to show that we never get inside the forbidden regions.

In Lemma 2, we have shown that for all $i > 0$; c_{k-i} and b_{k-i} satisfy the equation

$$c_{k-i} = \frac{P_i}{Q_{i-1}} - \frac{Q_i}{Q_{i-1}} (b_{k-i} - b_k).$$

The line of closest approach to the forbidden regions, or alternatively, the leftmost line is given by

$$c_{k-i} = \left(\frac{P_i}{Q_{i-1}}\right)_{\min} - \left(\frac{Q_i}{Q_{i-1}}\right)_{\min} (b_{k-i} - b_k)$$

Since we have $b_k \geq 0$, the worst case is then $b_k = 0$. Thus leftmost line is given by

$$c_{k-i} = \left(\frac{P_i}{Q_{i-1}}\right)_{\min} - \left(\frac{Q_i}{Q_{i-1}}\right)_{\min} b_{k-i}.$$

It is necessary to discuss each one of the three ranges separately. It is also necessary to discuss the cases of even and odd values of i, separately.

Consider the range I_1. As we noted earlier, we will consider the case $b_k = 0$, since this is the worst case.

Now for

$$b_k = 0 \qquad \frac{1}{4} \le c_k < \frac{25}{64} \qquad q_1 = 1$$

$$\therefore \quad \frac{1}{4} \le b_{k-1} < \frac{25}{64}$$

$$P_1 = 1, \; Q_1 = 1, \; \frac{P_1}{Q_0} = 1, \; \frac{Q_1}{Q_0} = 1,$$

$$\therefore \quad c_{k-1} = 1 - b_{k-1} \qquad \frac{1}{4} \le b_{k-1} < \frac{25}{64}$$

We call this segment P. This segment is clearly within the overlap regions, as shown in figure 7.

<u>Case (1)</u> i is odd - range I_1

For i = 1, we have seen that we don't go into the forbidden regions. For $i \ge 3$, we first get a lower bound on $\frac{P_i}{Q_i}$. We use a theorem[1] which states that odd ordered convergents approach the value of an infinite continued fraction from above (and even ordered convergents approach from below). Thus

$$\left(\frac{P_i}{Q_i}\right)_{i \text{ odd}} \ge \min_{u \in I_1} (u) = \frac{1}{2}$$

It is also clear that

$$\left(\frac{Q_i}{Q_{i-1}}\right)_{\substack{i \ge 3 \\ \text{odd}}} \ge \frac{1}{4} + \frac{1}{1 + \frac{1}{\frac{1}{4}}} = \frac{9}{20}.$$

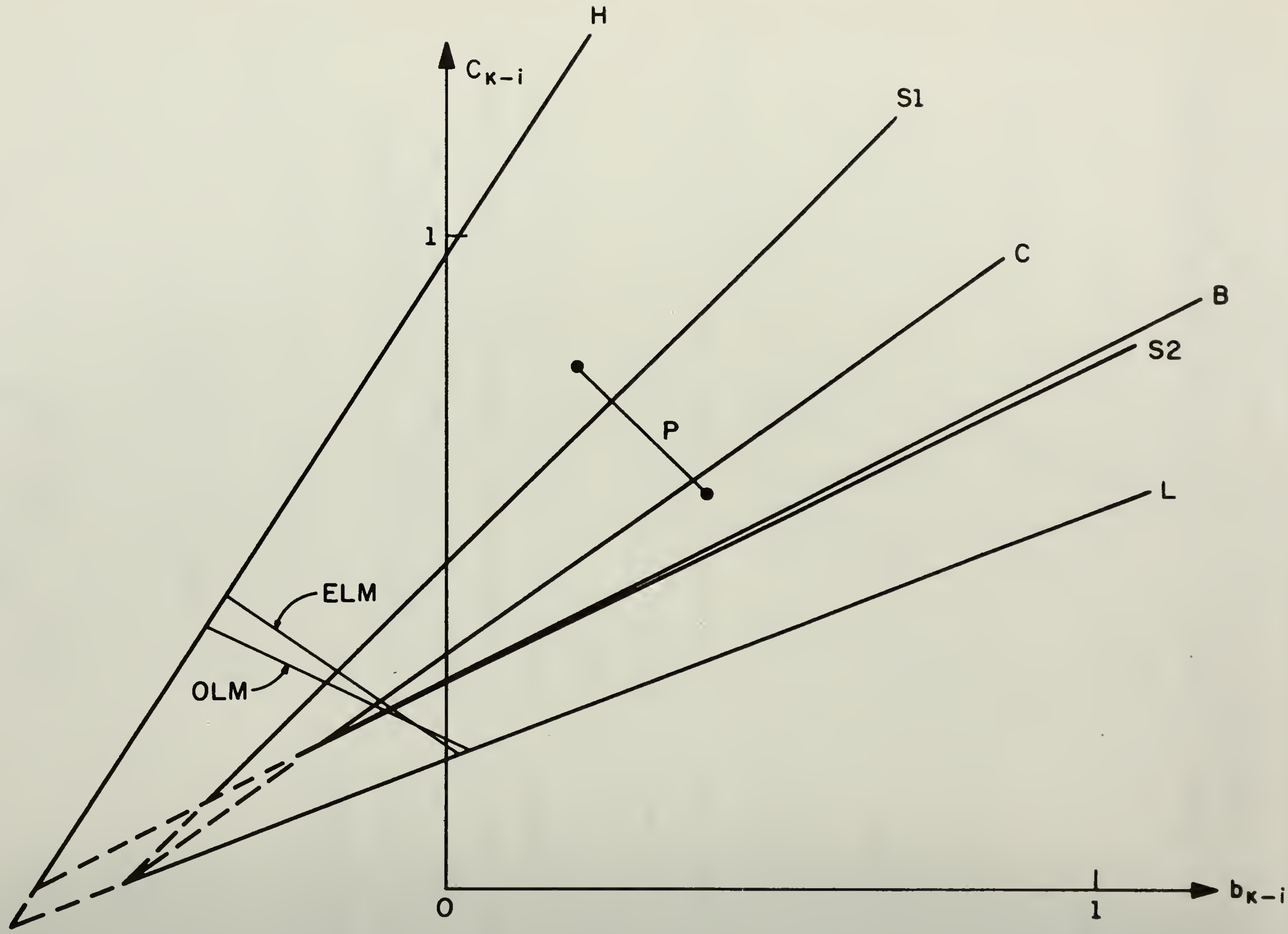

Figure 7

Thus

$$\left(\frac{P_i}{Q_{i-1}}\right)_{i \text{ odd} \geq 3} \geq \frac{9}{40}.$$

Therefore the leftmost line for odd i is given by

$c_{k-i} = \frac{9}{40} - \frac{9}{20} b_{k-i}$. We call this line OLM, and is shown in figure 7.

We can see that this line is well within the overlap regions.

Case (2) i even - range I_1

From figure 7, it is clear that $q_2 = \frac{1}{2}$ for the whole region I_1.

Then

$$\frac{P_2}{Q_2} = \frac{1}{3}.$$

Now again using the theorem quoted in the previous case,

$$\left(\frac{P_i}{Q_i}\right)_{i \text{ even} \geq 2} \geq \frac{1}{3}.$$

Next notice that

$$\left(\frac{Q_i}{Q_{i-1}}\right)_{i \text{ even}} \geq \frac{1}{4} + \cfrac{1}{1 + \cfrac{1}{\cfrac{1}{4} + \cdots \cfrac{1}{1}}} > \frac{1}{4} + \cfrac{1}{1 + \cfrac{1}{\cfrac{1}{4} + \cdots}}$$

The fraction on the right is an infinite continued fraction and is easily evaluated to be 0.640388. Thus, we have

$$\left(\frac{P_i}{Q_{i-1}}\right)_{i \text{ even} \geq 2} \geq 0.640388 * \frac{1}{3}.$$

Thus the leftmost line for even values of i, is given by

$$c_{k-i} = 0.2135 - 0.64\ b_{k-i}\ .$$

We call this line ELM. From figure 7, it is clear that this line is well within the overlap regions.

Now consider the second range I_2. For $b_k = 0$ and $\frac{25}{64} \leq c_k < \frac{9}{16}$, $q_1 = 1/2$. Therefore $P_1 = 1$, $Q_1 = \frac{1}{2}$, $\frac{P_1}{Q_0} = 1$, $\frac{Q_1}{Q_0} = \frac{1}{2}$. Thus $c_{k-1} = 1 - \frac{1}{2} b_{k-1}$ and $\frac{25}{128} \leq b_{k-1} < \frac{9}{32}$. This line segment is shown in figure 8 and is seen to be well within the overlap regions. We call this line segment P. It is also clear from figure 8 that only possible values of q_2 are $\frac{1}{4}$ and $\frac{1}{2}$. Corresponding values of $\frac{P_2}{Q_2}$ are $\frac{2}{9}$ and $\frac{2}{9}$ respectively.

<u>Case (3)</u> i odd - range I_2

We have
$$\left(\frac{P_i}{Q_i}\right)_{i\ \text{odd} \geq 3} \geq \min_{u \in I_2} (u) = 5/8.$$

As before,
$$\left(\frac{Q_i}{Q_{i-1}}\right)_{i\ \text{odd} \geq 3} \geq \frac{9}{20}.$$

$$\therefore \quad \left(\frac{P_i}{Q_{i-1}}\right)_{i\ \text{odd} \geq 3} \geq \frac{9}{32} = 0.281$$

Thus OLM is given by $c_{k-i} = 0.281 - 0.45\ b_{k-i}$. This line is clearly within limits as shown in figure 8.

<u>Case (4)</u> i even - range I_2

We have
$$\left(\frac{P_i}{Q_i}\right)_{i\ \text{even} \geq 2} \geq \frac{2}{9}\ ,$$

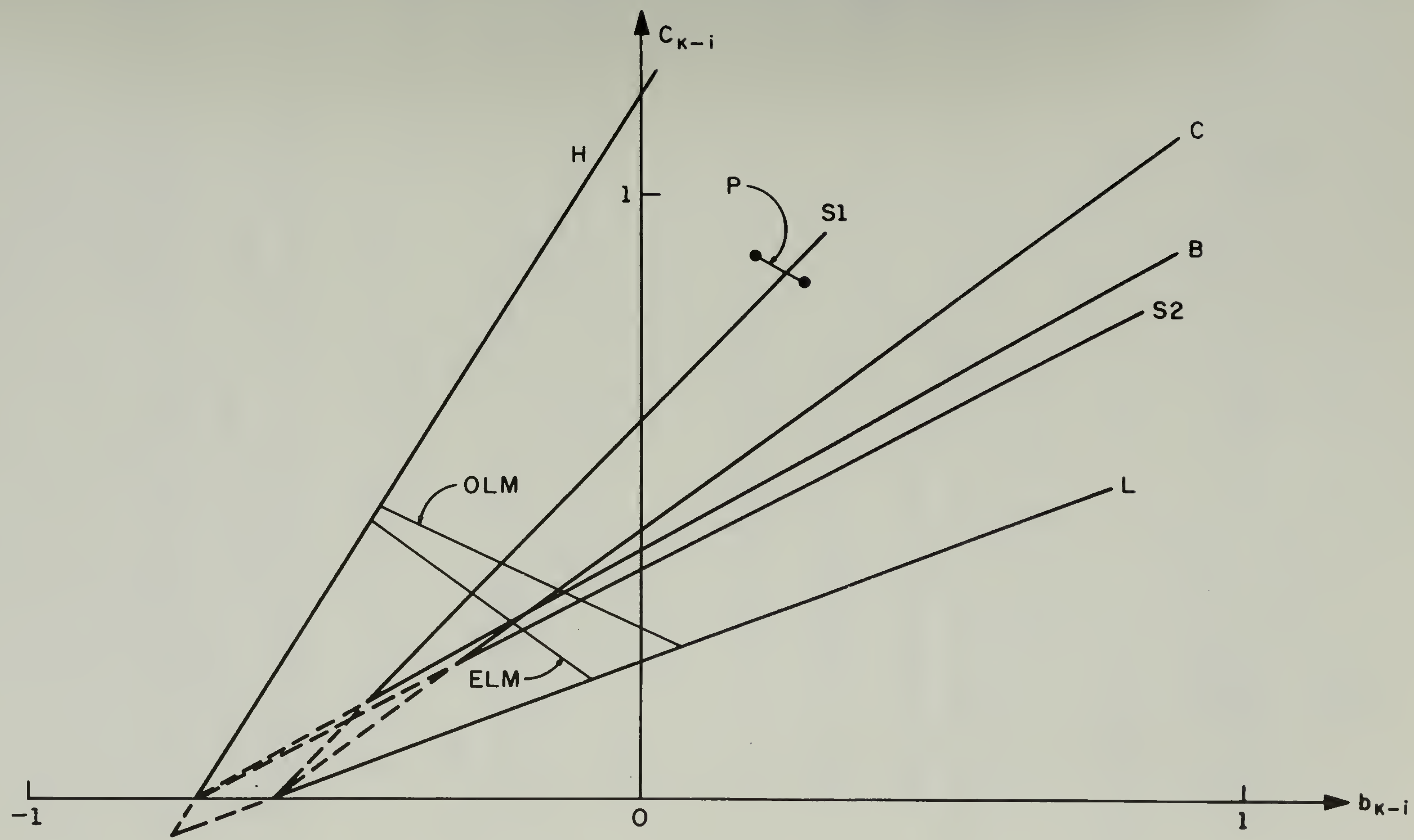

Figure 8

$$\left(\frac{Q_i}{Q_{i-1}}\right)_{i \text{ even } \geq 2} \geq 0.640388,$$

$$\therefore \quad \left(\frac{P_i}{Q_{i-1}}\right)_{i \text{ even } \geq 2} \geq 0.142.$$

Thus ELM is given by

$$c_{k-i} = 0.142 - 0.64\, b_{k-i}$$

which is also within limits as shown in figure 8.

Now consider the third range, I_3. For

$$b_k = 0 \quad \text{and} \quad \frac{9}{16} \leq c_k \leq 1, \qquad q_1 = \frac{1}{2}$$

$$\therefore \quad P_1 = 1,\ Q_1 = 1/2,\ \frac{P_1}{Q_0} = 1,\ \frac{Q_1}{Q_0} = 1/2$$

$$c_{k-1} = 1 - \frac{1}{2} b_{k-1} \qquad\qquad \frac{9}{32} \leq b_{k-1} \leq 1$$

is the line segment which is the reflection of the initial line segment. It is clear that this segment (called P) is within limits and also that $q_2 = 1/2$. Then

$$P_2 = 1/2,\ Q_2 = 5/4,\ \frac{P_2}{Q_1} = 1,\ \frac{Q_2}{Q_1} = 5/2.$$

Thus the second reflection is given by $c_{k-2} = 1 - 5/2\ b_{k-2}$. With the endpoints $(c_{k-2})_{c_k=3/4} = 0.595$ and $(c_{k-2})_{c_k=1} = 1.32$.

This line segment is once again well within limits and $q_3 = \frac{1}{2}$ or $\frac{1}{4}$.

<u>Case (5)</u> i odd - range I_3

$$\left(\frac{P_i}{Q_i}\right)_{i \text{ odd} \geq 3} \geq 3/4 ,$$

$$\left(\frac{Q_i}{Q_{i-1}}\right)_{i \text{ odd} \geq 3} \geq \frac{9}{20} .$$

Thus OLM is given by $c_{k-i} = 0.3375 - 0.45\, b_{k-i}$. This line is well within limits as shown in figure 9.

<u>Case (6)</u> i even - range I_3

We have

$$q_1 = 1/2,\ q_2 = 1/2,\ q_3 = \frac{1}{2} \text{ or } \frac{1}{4}$$

$$q_1 = \frac{1}{4} \text{ or } \frac{1}{2} \text{ or } 1$$

Minimum $$\frac{P_4}{Q_4} = \cfrac{1}{\cfrac{1}{2} + \cfrac{1}{\cfrac{1}{2} + \cfrac{1}{\cfrac{1}{2} + \cfrac{1}{\cfrac{1}{4}}}}} = \frac{26}{49} = 0.53.$$

Therefore $$\left(\frac{P_i}{Q_i}\right)_{i \text{ even} \geq 4} \geq 0.53$$

$$\left(\frac{Q_i}{Q_{i-1}}\right)_{i \text{ even}} \geq 0.64.$$

Therefore ELM is given by $c_{k-i} = 0.34 - 0.64\, b_{k-i}$. This line is within limits as shown in figure 9.

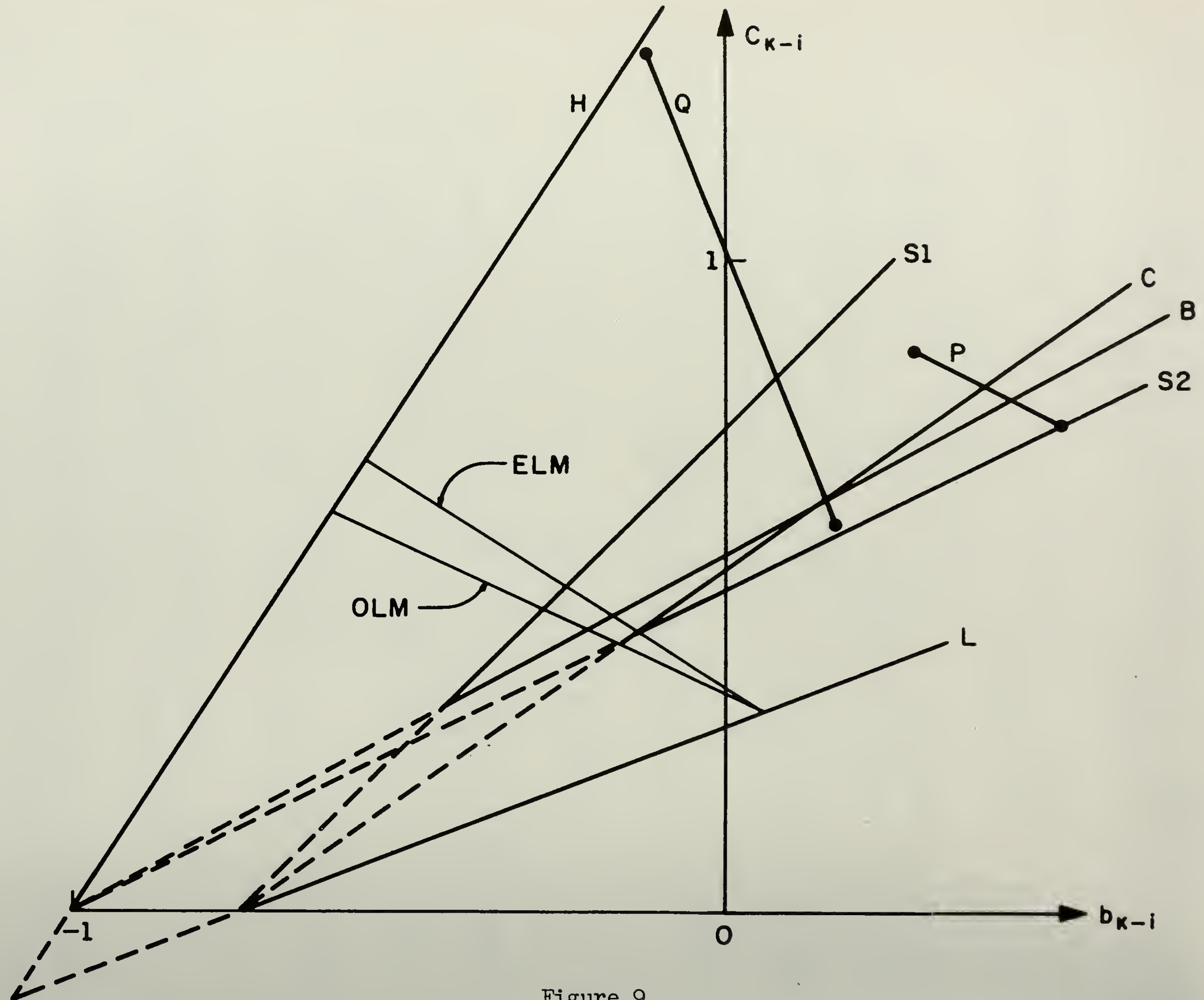

Figure 9

Thus we have shown that our selection procedure is consistent. We can now use theorem-1. We have a consistent method of expansion of u into a continued fraction (though u is unknown here, but it does not matter). $m = \frac{1}{4} > 0$. Clearly initial errors δ_1 and δ_2 are finite. Thus all conditions of theorem-1 are met and thus we conclude that algorithm QD is convergent.

We now prove an auxilliary result. This is that c_{k-i} and b_{k-i} are bounded above (we have just shown that they are bounded below). This fact can be used in hardware implementation of the algorithm in that a fixed point register can be used to store the quantities c_{k-i} and b_{k-i}. It will also be helpful in showing that the residual (to be defined shortly) approaches zero as i increases.

We know that for any $i > 0$, we have, $c_{k-i} = \frac{P_i}{Q_{i-1}} - \frac{Q_i}{Q_{i-1}} (b_{k-i} - b_k)$.

We also know that $c_{k-i} \leq 1.56\ b_{k-i} + 1.56$. The largest positive c_{k-i} will then be given by the intersection of these two lines with appropriate extremal values of the quantities involved. The intersection is given by,

$$c_{k-i} = \frac{\left(\frac{P_i}{Q_i}\right) + 1 + b_k}{\frac{1}{1.56} + \frac{Q_{i-1}}{Q_i}}$$

Therefore

$$(c_{k-i})_{max} = \frac{\left(\frac{P_i}{Q_i}\right)_{max} + 1 + b_k}{\frac{1}{1.56} + \frac{1}{\left(\frac{Q_i}{Q_{i-1}}\right)_{max}}}$$

For all three u-ranges, we have, $(q_1)_{min} = 1/2$ and since odd ordered convergents approach the root from above and also since all even convergents are smaller than all odd convergents, we have

$$\left(\frac{P_i}{Q_i}\right)_{max} = \frac{1}{\frac{1}{2}} = 2.$$

We also have,

$$\left(\frac{Q_i}{Q_{i-1}}\right)_{max} = 1 + \frac{1}{\frac{1}{4}} = 5$$

$$\therefore \quad c_{k-i\ max} \leq \frac{2 + 1 + b_k}{0.64 + 0.2}$$

$$= 3.57 + 1.2\ b_k$$

Thus for any given b_k, c_{k-i} is bounded above. Similarly taking the intersection of line (5.3) with line L. We can get a bound on b_{k-i}.

Now define residual at step i by

$$r_i = \left(\frac{P_i}{Q_i}\right)^2 + b_k \left(\frac{P_i}{Q_i}\right) - c_k \ .$$

Using equation (5.1), we have

$$r_i = (-1)^{i+1} \frac{c_{k-i}}{Q_i^2} \ .$$

From the recursion $Q_i = q_i Q_{i-1} + Q_{i-2}$, it is clear that $Q_i > Q_{i-2}$. Therefore Q_{2j} and Q_{2j+1} form two increasing sequences as j increases. Since c_{k-i} is bounded, we must have $r_i \to 0$ as $i \to \infty$.

6. CONCLUSIONS

To make practical use of the continued fraction representation of numbers, we need to develop algorithms for many more functions. This is an open area of research. What has been shown in this paper is just a beginning.

We now compare the algorithm QD of this paper with standard methods of solving for the root u of the quadratic

$$x^2 + b_k x - c_k = 0 = (x-u)\ (x+v).$$

Then

$$u = \frac{-b_1 + \sqrt{(b_k^2 + 4\ c_k)}}{2}$$

Neglecting operations like shift and add, we thus need a sqaure root and a multiplication to be carried out by standard methods.

In the IBM-360[1] square root subroutine, use is made of the Newton-Ralphson iterative technique. For a single precision result two iterative steps need be carried out. Each of these steps essentially amounts to one division. Finding a suitable initial approximation also amounts to a division to be carried out. Thus all in all the solving for u essentially amounts to one multiplication and three division operations. Each of these operations is about ten times slower than one addition in the IBM 360/75. Thus approximately forty additions yield the root u of the quadratic. Our algorithm QD compares favorably with

this because on the average it requires less than forty iterations to get a single precision result. Further, the author believes that it is possible to improve the selection procedure of algorithm QD, so as to improve the rate of convergence of that algorithm.

A detailed study of the convergence behavior of this algorithm is desired. Such a study may have to be experimental. A mathematical analysis of the algorithm is made difficult by the fact that b_{k-i}, c_{k-i} at step $i + 1$ depend not only on the present b_{k-i}, c_{k-i} values, but also on the immediately preceeding ones. In other words, the next state is a function not only of the present state but also of the immediately past state.

LIST OF REFERENCES

[1] Wall, H., "Analytic Theory of Continued Fractions," Van Nostrand, New York, 1950.

[2] Robertson, J. E., Private communication, 1971.

[3] DeLugish, B. G., "A Class of Algorithms for Automatic Evaluation of Certain Elementary Functions in a Binary Computer," Report 399, Department of Computer Science, University of Illinois, Urbana, Illinois, 1970.

[4] IBM System/360 Operating System FORTRAN IV Library, Mathematical and Service Subprograms, Form GC28-6818-0.

[5] Khinchine, A. Ya., "Continued Fractions," Moscow-Leningrad State Publishing House, 1949.

[6] Khovanskii, A. N., "The Application of Continued Fractions," P. Noordhoff N. V. - Groningen - The Netherlands.

[7] Buchholz, W., "Choosing a Number Base," in "Planning a Computer System," Ed. Buchholz, W., McGraw Hill, 1962.

[8] Robertson, J. E., Class notes for C. S. 394, University of Illinois at Urbana-Champaign.

[9] Bracha, A., Private communication, 1971.

APPENDIX I

A FORTRAN PROGRAM IMPLEMENTING THE ALGORITHM QD

```
FORTRAN IV G LEVEL  18                    MAIN                    DATE = 72160

          C    THIS IS A FORTRAN PROGRAM THAT SOLVES FOR THE SMALLER
          C    POSITIVE ROOT OF THE QUADRATIC X**2+BKNP1 * X - CKNP1=0:
          C    (USING ALGORITHM QD) TO TEN DIGIT ACCURACY.
                 IMPLICIT REAL*8(A-H,O-Z)
                 REAL*8 K1,K2
          C    SETTING UP TEST DATA
                 DIMENSION CK(4),BK(4)
                 CK(1)=0.5
                 BK(1)=0.0
                 CK(2)=.625
                 BK(2)=0.0
                 CK(3)=1.25
                 BK(3)=.75
                 CK(4)=2.125
                 BK(4)=1.5
                 DO 200 J=1,4
                 CKNP1=CK(J)
                 BKNP1=BK(J)
          C    STEP QD_0
                 IF(CKNP1-.5*BKNP1.LT..25.OR.CKNP1-BKNP1.GT.1..OR.
               -  BKNP1.LT.0.)   GO TO 200
          C    FIND THE ROOT BY THE STANDARD METHOD FOR COMPARISON
                 R1=.5D0*(-BKNP1+DSQRT(BKNP1**2+4.*CKNP1))
                 WRITE(6,75)   CKNP1,BKNP1
           75    FORMAT('1','*******************************************
               -*****************************************************'
               -'*************'/'****CK=',D25.16,'  BK=',D15.7,
               -'**************'/' STEP',T11,'C(K-N)',T25,'B(K-N)',T40,
               -' (N)',T58,'ROOT',T79,'ERROR')
          C    STEP QD_1
          250    I=1
                 IF((CKNP1-.625D0*BKNP1).LT.25.D0/64.D0) GO TO 101
                 SN=.5D0
                 IF((CKNP1-.75D0*BKNP1).LT..5625D0) GO TO 102
                 K1=.75D0
                 K2=.5D0
                 GO TO  2
          101    SN=1.D0
                 K1=.5D0
                 K2=.3125D0
                 GO TO  2
          102    K1=.625D0
                 K2=.375D0
```

```
C     STEP QD_2 AND STEP QD_3
 2        BKN=SN*CKNP1
          CKN=1.D0-SN*(BKN-BKNP1)
          SN1=SN
          P=1.D0
          Q=SN
          R=P/Q
          PM1=0.
          QM1=1.D0
C     STEP QD_4
 1000     TEMP=CKN-.5D0*BKN
          IF(TEMP.LE.K2) GO TO 1
          IF(CKN-BKN.GT.K1 ) GO TO 25
 500      SN=.5D0
          GO TO 100
 1        SN=1.D0
          GO TO 100
 25       SN=.25D0
 100      BKNP2=BKNP1
          ERROR=R-R1
          WRITE(6,85)  I,CKN,BKN,SN1,R,ERROR
 85       FORMAT(' ',I4, 3D14.5,D25.16,D14.5)
          BKNP1=BKN
          CKNP2=CKNP1
          CKNP1=CKN
          PM2=PM1
          PM1=P
          QM2=QM1
          QM1=Q
          I=I+1
C     STEP QD_5
 10       BKN=SN*CKNP1-SN1*CKNP2+BKNP2
          CKN=SN*(BKNP1-BKN)+CKNP2
          P=SN*PM1+PM2
          Q=SN*QM1+QM2
          R=P/Q
          SN1=SN
          T=DABS(R-R1)
C     STEP QD_6
          IF(T.GT..5D-10.AND.I.LT.050)     GO TO 1000
 200      CONTINUE
          END
```

APPENDIX II

COMPUTER PRINTOUTS OF A FEW PROBLEMS SOLVED USING THE PROGRAM IN APPENDIX I

***CK= 0.5000000000000000D 00 ***BK= 0.0

TEP	C(K-N)	B(K-N)	(N)	ROOT	ERROR
1	0.87500D 00	0.25000D 00	0.50000D 00	0.2000000000000000D 01	0.12929D 01
2	0.53125D 00	0.18750D 00	0.50000D 00	0.4000000000000000D 00	-0.30711D 00
3	0.92969D 00	0.78125D-01	0.50000D 00	0.1111111111111111D 01	0.40400D 00
4	0.51221D 00	0.15430D 00	0.25000D 00	0.5306122448979592D 00	-0.17649D 00
5	0.95593D 00	0.10181D 00	0.50000D 00	0.8760330578512397D 00	0.16893D 00
6	0.50336D 00	0.13718D 00	0.25000D 00	0.6120857699805068D 00	-0.95021D-01
7	0.96727D 00	0.11451D 00	0.50000D 00	0.7846049966239028D 00	0.77493D-01
8	0.50016D 00	0.12731D 00	0.25000D 00	0.6578334825425246D 00	-0.49273D-01
9	0.96954D 00	0.12277D 00	0.50000D 00	0.7439912809040326D 00	0.36884D-01
10	0.50095D 00	0.11961D 00	0.25000D 00	0.6820150371097838D 00	-0.25092D-01
11	0.96391D 00	0.13086D 00	0.50000D 00	0.7246942091607673D 00	0.17787D-01
12	0.50614D 00	0.11012D 00	0.25000D 00	0.6943896827544342D 00	-0.12717D-01
13	0.94750D 00	0.14295D 00	0.50000D 00	0.7156743505147123D 00	0.85681D-02
14	0.51840D 00	0.93922D-01	0.25000D 00	0.7006173609260477D 00	-0.64894D-02
15	0.91182D 00	0.16528D 00	0.50000D 00	0.7111675161337351D 00	0.40607D-02
16	0.54405D 00	0.62673D-01	0.25000D 00	0.7037251616252647D 00	-0.33816D-02
17	0.83849D 00	0.20934D 00	0.50000D 00	0.7089502804016092D 00	0.18435D-02
18	0.59631D 00	0.27721D-02	0.25000D 00	0.7052695146847250D 00	-0.18373D-02
19	0.68969D 00	0.29788D 00	0.50000D 00	0.7078562816902566D 00	0.74950D-03
20	0.72177D 00	0.46964D-01	0.50000D 00	0.7064512903912505D 00	-0.65549D-03
21	0.66806D 00	0.13348D 00	0.25000D 00	0.7075544468213571D 00	0.44767D-03
22	0.68823D 00	0.20055D 00	0.50000D 00	0.7068570649965219D 00	-0.24972D-03
23	0.69655D 00	0.14357D 00	0.50000D 00	0.7072722570461180D 00	0.16548D-03
24	0.65767D 00	0.20471D 00	0.50000D 00	0.7070156219758471D 00	-0.91159D-04
25	0.73684D 00	0.12413D 00	0.50000D 00	0.7071707220609022D 00	0.63941D-04
26	0.59758D 00	0.24429D 00	0.50000D 00	0.7070756787380341D 00	-0.31102D-04
27	0.83173D 00	0.54500D-01	0.50000D 00	0.7071334333797055D 00	0.26652D-04
28	0.57285D 00	0.15343D 00	0.25000D 00	0.7070896336239281D 00	-0.17148D-04
29	0.84195D 00	0.13299D 00	0.50000D 00	0.7071184939918026D 00	0.11719D-04
30	0.58672D 00	0.77436D-01	0.25000D 00	0.7070973775671612D 00	-0.94036D-05
31	0.77277D 00	0.21587D 00	0.50000D 00	0.7071117862014359D 00	0.50050D-05
32	0.60940D 00	0.17052D 00	0.50000D 00	0.7071037210922786D 00	-0.30601D-05
33	0.79094D 00	0.13418D 00	0.50000D 00	0.7071088651551835D 00	0.20840D-05
34	0.62705D 00	0.63554D-01	0.25000D 00	0.7071050409407309D 00	-0.17402D-05
35	0.69773D 00	0.24997D 00	0.50000D 00	0.7071076123890264D 00	0.83120D-06
36	0.70259D 00	0.98891D-01	0.50000D 00	0.7071061539525153D 00	-0.62723D-06
37	0.62097D 00	0.25241D 00	0.50000D 00	0.7071070784249744D 00	0.29724D-06
38	0.79976D 00	0.58080D-01	0.50000D 00	0.7071065290706520D 00	-0.25212D-06
39	0.60003D 00	0.14186D 00	0.25000D 00	0.7071069490469058D 00	0.16786D-06
40	0.79161D 00	0.15815D 00	0.50000D 00	0.7071066746897174D 00	-0.10650D-06
41	0.62963D 00	0.39749D-01	0.25000D 00	0.7071068763474219D 00	0.95161D-07
42	0.67395D 00	0.27506D 00	0.50000D 00	0.7071067393267207D 00	-0.41860D-07
43	0.73620D 00	0.61912D-01	0.50000D 00	0.7071068163062729D 00	0.35120D-07
44	0.65890D 00	0.12214D 00	0.25000D 00	0.7071067564123190D 00	-0.24774D-07
45	0.69362D 00	0.20731D 00	0.50000D 00	0.7071067947275209D 00	0.13541D-07
46	0.69230D 00	0.13950D 00	0.50000D 00	0.7071067721079980D 00	-0.90785D-08
47	0.65992D 00	0.20690D 00	0.50000D 00	0.7071067861549251D 00	0.49684D-08
48	0.73472D 00	0.12306D 00	0.50000D 00	0.7071067776911172D 00	-0.34954D-08
49	0.59929D 00	0.24431D 00	0.50000D 00	0.7071067828866536D 00	0.17003D-08

****

****CK= 0.62500000000000000D 00 BK= 0.0

STEP	C(K-N)	B(K-N)	(N)	ROOT	ERROR
1	0.84375D 00	0.31250D 00	0.50000D 00	0.20000000000000000D 01	0.12094D 01
2	0.72656D 00	0.10938D 00	0.50000D 00	0.40000000000000000D 00	-0.39057D 00
3	0.77148D 00	0.25391D 00	0.50000D 00	0.11111111111111111D 01	0.32054D 00
4	0.78760D 00	0.13184D 00	0.50000D 00	0.62068965517241380D 00	-0.16988D 00
5	0.70642D 00	0.26196D 00	0.50000D 00	0.89230769230769230D 00	0.10174D 00
6	0.87296D 00	0.91248D-01	0.50000D 00	0.71823204419889500D 00	-0.72337D-01
7	0.69748D 00	0.12699D 00	0.25000D 00	0.84736091298145500D 00	0.56791D-01
8	0.82558D 00	0.27175D 00	0.50000D 00	0.76035365286179620D 00	-0.30216D-01
9	0.71284D 00	0.19104D 00	0.50000D 00	0.80961033716939230D 00	0.19041D-01
10	0.83940D 00	0.16538D 00	0.50000D 00	0.77836002657022660D 00	-0.12209D-01
11	0.66863D 00	0.25382D 00	0.50000D 00	0.79691855759719430D 00	0.63491D-02
12	0.92506D 00	0.80497D-01	0.50000D 00	0.78543120481514900D 00	-0.51382D-02
13	0.65106D 00	0.15077D 00	0.25000D 00	0.79403088702055320D 00	0.35115D-02
14	0.91306D 00	0.17476D 00	0.50000D 00	0.78833748380179750D 00	-0.22319D-02
15	0.59756D 00	0.23177D 00	0.50000D 00	0.79163062279363980D 00	0.10612D-02
16	0.87905D 00	0.31579D 00	0.10000D 01	0.79011444232308120D 00	-0.45497D-03
17	0.69358D 00	0.12374D 00	0.50000D 00	0.79090140530203840D 00	0.33199D-03
18	0.82939D 00	0.22305D 00	0.50000D 00	0.79038359737031420D 00	-0.18582D-03
19	0.70928D 00	0.19164D 00	0.50000D 00	0.79068274284602420D 00	0.11333D-03
20	0.84371D 00	0.16300D 00	0.50000D 00	0.79049483945246640D 00	-0.74576D-04
21	0.66135D 00	0.25886D 00	0.50000D 00	0.79060720272067360D 00	0.37788D-04
22	0.93723D 00	0.71819D-01	0.50000D 00	0.79053792248638200D 00	-0.31493D-04
23	0.63868D 00	0.16249D 00	0.25000D 00	0.79059016971308910D 00	0.20755D-04
24	0.94005D 00	0.15685D 00	0.50000D 00	0.79055553280297060D 00	-0.13882D-04
25	0.65836D 00	0.79162D-01	0.25000D 00	0.79058079941976470D 00	0.11384D-0
26	0.85363D 00	0.25102D 00	0.50000D 00	0.79056351787112830D 00	-0.58972D-0
27	0.69597D 00	0.17580D 00	0.50000D 00	0.79057316666309160D 00	0.37516D-0
28	0.85543D 00	0.17219D 00	0.50000D 00	0.79056700536508680D 00	-0.24097D-0
29	0.65429D 00	0.25553D 00	0.50000D 00	0.79057064750263450D 00	0.12325D-0
30	0.94739D 00	0.71618D-01	0.50000D 00	0.79056838733195180D 00	-0.10277D-0
31	0.63089D 00	0.16523D 00	0.25000D 00	0.79057008743160880D 00	0.67239D-0
32	0.95489D 00	0.15022D 00	0.50000D 00	0.79056895737560230D 00	-0.45767D-0
33	0.64632D 00	0.88507D-01	0.25000D 00	0.79056978055140390D 00	0.36551D-0
34	0.88182D 00	0.23465D 00	0.50000D 00	0.79056921684824460D 00	-0.19819D-0
35	0.66052D 00	0.20626D 00	0.50000D 00	0.79056953121757720D 00	0.11618D-0
36	0.92295D 00	0.12400D 00	0.50000D 00	0.79056933036946130D 00	-0.84673D-0
37	0.66483D 00	0.10674D 00	0.25000D 00	0.79056947957056660D 00	0.64528D-0
38	0.86348D 00	0.22568D 00	0.50000D 00	0.79056937917333720D 00	-0.35869D-0
39	0.67464D 00	0.20606D 00	0.50000D 00	0.79056943607853640D 00	0.21036D-0
40	0.90088D 00	0.13126D 00	0.50000D 00	0.79056939999639040D 00	-0.15046D-0
41	0.68396D 00	0.93962D-01	0.25000D 00	0.79056942689125670D 00	0.11849D-0
42	0.82385D 00	0.24802D 00	0.50000D 00	0.79056940835221350D 00	-0.61399D-0
43	0.72602D 00	0.16391D 00	0.50000D 00	0.79056941910608350D 00	0.40640D-0
44	0.80625D 00	0.19911D 00	0.50000D 00	0.79056941261331550D 00	-0.24288D-0
45	0.72356D 00	0.20402D 00	0.50000D 00	0.79056941647444200D 00	0.14323D-0
46	0.82938D 00	0.15776D 00	0.50000D 00	0.79056941408631080D 00	-0.95579D-0
47	0.67398D 00	0.25693D 00	0.50000D 00	0.79056941552901240D 00	0.48692D-0
48	0.91781D 00	0.80062D-01	0.50000D 00	0.79056941464472970D 00	-0.39737D-0
49	0.65665D 00	0.14939D 00	0.25000D 00	0.79056941531315040D 00	0.27106D-0

CK= 0.1250000000000000D 01**BK= 0.7500000D 00********

TEP	C(K-N)	B(K-N)	Q(N)	ROOT	ERROR
1	0.10625D 01	0.62500D 00	0.50000D 00	0.2000000000000000D 01	0.11958D 01
2	0.12344D 01	0.65625D 00	0.50000D 00	0.4000000000000000D 00	-0.40425D 00
3	0.10352D 01	0.71094D 00	0.50000D 00	0.1111111111111111D 01	0.30686D 00
4	0.13115D 01	0.55664D 00	0.50000D 00	0.6206896551724138D 00	-0.18356D 00
5	0.10440D 01	0.52124D 00	0.25000D 00	0.9702970297029703D 00	0.16605D 00
6	0.11968D 01	0.75076D 00	0.50000D 00	0.7267267267267267D 00	-0.77521D-01
7	0.11206D 01	0.59762D 00	0.50000D 00	0.8602442333755617D 00	0.55997D-01
8	0.11392D 01	0.71267D 00	0.50000D 00	0.7742870952150797D 00	-0.29961D-01
9	0.11734D 01	0.60695D 00	0.50000D 00	0.8247956946322300D 00	0.20548D-01
10	0.10778D 01	0.72977D 00	0.50000D 00	0.7933498833972767D 00	-0.10898D-01
11	0.12588D 01	0.55914D 00	0.50000D 00	0.8122658193699229D 00	0.80182D-02
12	0.94727D 00	0.82024D 00	0.50000D 00	0.8006425294396357D 00	-0.36051D-02
13	0.14672D 01	0.40339D 00	0.50000D 00	0.8076935315964851D 00	0.34459D-02
14	0.86976D 00	0.71341D 00	0.25000D 00	0.8023428019569390D 00	-0.19048D-02
15	0.15882D 01	0.47148D 00	0.50000D 00	0.8058667910153114D 00	0.16191D-02
16	0.81874D 00	0.67556D 00	0.25000D 00	0.8032872667310934D 00	-0.96037D-03
17	0.13705D 01	0.89318D 00	0.10000D 01	0.8046218712141268D 00	0.37423D-03
18	0.99430D 00	0.54208D 00	0.50000D 00	0.8039663745542504D 00	-0.28127D-03
19	0.12890D 01	0.70507D 00	0.50000D 00	0.8044059193307969D 00	0.15828D-03
20	0.10021D 01	0.68945D 00	0.50000D 00	0.8041560140896818D 00	-0.91627D-04
21	0.13530D 01	0.56160D 00	0.50000D 00	0.8043142372740333D 00	0.66596D-04
22	0.10108D 01	0.52664D 00	0.25000D 00	0.8041962239148208D 00	-0.51418D-04
23	0.12519D 01	0.72878D 00	0.50000D 00	0.8042753283763042D 00	0.27687D-04
24	0.10517D 01	0.64716D 00	0.50000D 00	0.8042303383640959D 00	-0.17303D-04
25	0.12611D 01	0.62867D 00	0.50000D 00	0.8042588184792969D 00	0.11177D-04
26	0.99004D 00	0.75190D 00	0.50000D 00	0.8042418787166492D 00	-0.57628D-05
27	0.13905D 01	0.49312D 00	0.50000D 00	0.8042523549673442D 00	0.47135D-05
28	0.96219D 00	0.60451D 00	0.25000D 00	0.8042444638694082D 00	-0.31776D-05
29	0.13795D 01	0.62658D 00	0.50000D 00	0.8042497016995117D 00	0.20602D-05
30	0.10018D 01	0.46829D 00	0.25000D 00	0.8042458833799619D 00	-0.17581D-05
31	0.12223D 01	0.78259D 00	0.50000D 00	0.8042484964559241D 00	0.85495D-06
32	0.11038D 01	0.57857D 00	0.50000D 00	0.8042470362844190D 00	-0.60322D-06
33	0.11500D 01	0.72331D 00	0.50000D 00	0.8042479696379751D 00	0.32813D-06
34	0.11646D 01	0.60167D 00	0.50000D 00	0.8042474191332588D 00	-0.22232D-06
35	0.10855D 01	0.73062D 00	0.50000D 00	0.8042477608075778D 00	0.11930D-06
36	0.12488D 01	0.56212D 00	0.50000D 00	0.8042475548824454D 00	-0.86625D-07
37	0.96040D 00	0.81230D 00	0.50000D 00	0.8042476812651111D 00	0.39758D-07
38	0.14460D 01	0.41790D 00	0.50000D 00	0.8042476045401739D 00	-0.36967D-07
39	0.89147D 00	0.69361D 00	0.25000D 00	0.8042476627473913D 00	0.21240D-07
40	0.15418D 01	0.50212D 00	0.50000D 00	0.8042476244005853D 00	-0.17106D-07
41	0.85867D 00	0.63332D 00	0.25000D 00	0.8042476524656848D 00	0.10959D-07
42	0.15854D 01	0.54601D 00	0.50000D 00	0.8042476333243237D 00	-0.81828D-08
43	0.84509D 00	0.60035D 00	0.25000D 00	0.8042476470642050D 00	0.55571D-08
44	0.15995D 01	0.57220D 00	0.50000D 00	0.8042476375422312D 00	-0.39648D-08
45	0.84372D 00	0.57768D 00	0.25000D 00	0.8042476443098947D 00	0.28016D-08
46	0.15913D 01	0.59417D 00	0.50000D 00	0.8042476395835144D 00	-0.19236D-08
47	0.85335D 00	0.55364D 00	0.25000D 00	0.8042476429245090D 00	0.14174D-08
48	0.15565D 01	0.62329D 00	0.50000D 00	0.8042476405826919D 00	-0.92438D-09
49	0.88071D 00	0.51583D 00	0.25000D 00	0.8042476422342221D 00	0.72715D-09

****CK= 0.2125000000000000D 01 **BK= 0.1500000D 01 ***

STEP	C(K-N)	B(K-N)	(N)	ROOT	ERROR
1	0.12188D 01	0.10625D 01	0.50000D 00	0.2000000000000000D 01	0.11106D 01
2	0.21326D 01	0.10469D 01	0.50000D 00	0.4000000000000000D 00	-0.48936D 00
3	0.12339D 01	0.98633D 00	0.25000D 00	0.1384615384615384D 01	0.49526D 00
4	0.20607D 01	0.11306D 01	0.50000D 00	0.6415094339622641D 00	-0.24785D 00
5	0.12954D 01	0.88455D 00	0.25000D 00	0.1133757961783439D 01	0.24440D 00
6	0.18714D 01	0.12631D 01	0.50000D 00	0.7745266781411360D 00	-0.11483D 00
7	0.13407D 01	0.11725D 01	0.50000D 00	0.9611248966087676D 00	0.71765D-01
8	0.19587D 01	0.99782D 00	0.50000D 00	0.8383809793376734D 00	-0.50979D-01
9	0.13422D 01	0.99186D 00	0.25000D 00	0.9282847406285498D 00	0.38925D-0
10	0.18650D 01	0.11792D 01	0.50000D 00	0.8670090911283127D 00	-0.22351D-01
11	0.13052D 01	0.12533D 01	0.50000D 00	0.9013352564986372D 00	0.11976D-0
12	0.20420D 01	0.89930D 00	0.50000D 00	0.8794495800741837D 00	-0.99101D-0
13	0.12522D 01	0.11112D 01	0.25000D 00	0.8957226235784547D 00	0.63630D-0
14	0.20902D 01	0.10149D 01	0.50000D 00	0.8847821740249764D 00	-0.45775D-0
15	0.12540D 01	0.10077D 01	0.25000D 00	0.8927017013519821D 00	0.33421D-0
16	0.20344D 01	0.11193D 01	0.50000D 00	0.8872500906040079D 00	-0.21095D-0
17	0.13115D 01	0.88925D 00	0.25000D 00	0.8911409740678774D 00	0.17813D-0
18	0.18457D 01	0.12665D 01	0.50000D 00	0.8884326225090059D 00	-0.92701D-0
19	0.13666D 01	0.11563D 01	0.50000D 00	0.8899186255298404D 00	0.55399D-0
20	0.19104D 01	0.10270D 01	0.50000D 00	0.8889623347093139D 00	-0.39730D-0
21	0.13857D 01	0.95064D 00	0.25000D 00	0.8896703506563584D 00	0.31072D-0
22	0.17646D 01	0.12422D 01	0.50000D 00	0.8891924354726253D 00	-0.16720D-0
23	0.14368D 01	0.11401D 01	0.50000D 00	0.8894625591903061D 00	0.10293D-0
24	0.17955D 01	0.10783D 01	0.50000D 00	0.8892910513862472D 00	-0.68580D-0
25	0.13162D 01	0.13195D 01	0.50000D 00	0.8893923492823454D 00	0.33218D-0
26	0.20360D 01	0.83859D 00	0.50000D 00	0.8893298195298123D 00	-0.29812D-0
27	0.12332D 01	0.11704D 01	0.25000D 00	0.8893772735943041D 00	0.17643D-0
28	0.21481D 01	0.94619D 00	0.50000D 00	0.8893457601264337D 00	-0.13871D-0
29	0.11970D 01	0.10908D 01	0.25000D 00	0.8893687271771581D 00	0.90961D-0
30	0.21897D 01	0.10077D 01	0.50000D 00	0.8893530061876820D 00	-0.66249D-0
31	0.11890D 01	0.10397D 01	0.25000D 00	0.8893642660373477D 00	0.46350D-0
32	0.21821D 01	0.10548D 01	0.50000D 00	0.8893564494011885D 00	-0.31817D-0
33	0.12050D 01	0.99074D 00	0.25000D 00	0.8893619979563506D 00	0.23669D-0
34	0.21216D 01	0.11118D 01	0.50000D 00	0.8893581200056426D 00	-0.15111D-0
35	0.12533D 01	0.91862D 00	0.25000D 00	0.8893603602955444D 00	0.12292D-0
36	0.19769D 01	0.12081D 01	0.50000D 00	0.8893589387524639D 00	-0.69232D-0
37	0.13588D 01	0.78617D 00	0.25000D 00	0.8893602935102698D 00	0.66243D-0
38	0.16733D 01	0.13932D 01	0.50000D 00	0.8893593419891340D 00	-0.28909D-0
39	0.15837D 01	0.94344D 00	0.50000D 00	0.8893598591406444D 00	0.22807D-0
40	0.14709D 01	0.13484D 01	0.50000D 00	0.8893595249867758D 00	-0.10609D-0
41	0.18144D 01	0.88701D 00	0.50000D 00	0.8893597207991531D 00	0.89724D-0
42	0.14260D 01	0.10666D 01	0.25000D 00	0.8893595703973583D 00	-0.60678D-0
43	0.17745D 01	0.11464D 01	0.50000D 00	0.8893596681239604D 00	0.37048D-0
44	0.13787D 01	0.12409D 01	0.50000D 00	0.8893596111012027D 00	-0.19974D-0
45	0.19207D 01	0.94849D 00	0.50000D 00	0.8893596467353234D 00	0.15660D-0
46	0.13579D 01	0.10317D 01	0.25000D 00	0.8893596200081348D 00	-0.11067D-0
47	0.18629D 01	0.11473D 01	0.50000D 00	0.8893596378251949D 00	0.67497D-0
48	0.12895D 01	0.12842D 01	0.50000D 00	0.8893596276434959D 00	-0.34320D-0
49	0.20747D 01	0.86055D 00	0.50000D 00	0.8893596340738807D 00	0.29984D-0

IBLIOGRAPHIC DATA HEET	1. Report No. UIUCDCS-R-72-525	2.	3. Recipient's Accession No.

Title and Subtitle

An Algorithm for the Solution of a Quadratic Equation using Continued Fractions

5. Report Date: June, 1972

6. Date of approval

Author(s)

Kishor Shridharbhai Tivedi

8. Performing Organization Rept. No.

Performing Organization Name and Address

Department of Computer Science
University of Illinois
Urbana, Illinois 61801

10. Project/Task/Work Unit No.

11. Contract/Grant No.

NSF GJ-813

2. Sponsoring Organization Name and Address

National Science Foundation
Washington, D. C.

13. Type of Report & Period Covered

Research

14.

5. Supplementary Notes

6. Abstracts

This is an effort to investigate representations of numbers other than positional notation for computer arithmetic. Using continued fraction representation of numbers, an algorithm to solve a limited class of quadratics has been developed. This algorithm is suitable for hardware implementation and is reasonably efficient. Feasibility of constructing an arithmetic unit with continued fraction representation depends on discovery of many more such useful algorithms which can share the same hardware.

7. Key Words and Document Analysis. 17a. Descriptors

Continued fraction,
Computer arithmetic,
Quadratic equation,
Square root algorithm,
Floating point,

7b. Identifiers/Open-Ended Terms

7c. COSATI Field/Group

8. Availability Statement	19. Security Class (This Report)	21. No. of Pages
Unlimited Release	UNCLASSIFIED	62
	20. Security Class (This Page) UNCLASSIFIED	22. Price